Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 00.xxx/ACCESS.2024.0000

Deep Learning-Based Image Watermarking using Catalan Transform and Non-Negative Matrix Factorization

MD. FAHIM HOSSAIN SAIKAT[1], MD. AL-MAMUN PROVATH[1] KAUSHIK DEB[1], PRANAB KUMAR DHAR[1], AND TETSUYA SHIMAMURA[2], (Senior Member, IEEE)

[1]Department of Computer Science and Engineering, Chittagong University of Engineering & Technology (CUET), Chattogram 4349, Bangladesh
[2]Department of Information and Computer Sciences, Saitama University, 255 Shimo-Okubo, Sakura-ku, Saitama 338-8570, Japan

Corresponding author: Kaushik Deb(debkaushik99@cuet.ac.bd)

ABSTRACT

In the digital age, embedding imperceptible data into images for authenticating ownership and content integrity through watermarking has gained immense importance. Existing watermarking methods struggle to balance imperceptibility, robustness, and security, resulting in suboptimal peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), normalized correlation (NC) values, along with higher time complexity. To overcome these constraints, this study proposes a new method for image watermarking that uses deep learning and operates within the YIQ color space. The main contribution of the proposed method is its new combination of the Catalan transform (CT) and non-negative matrix factorization (NMF) with an artificial neural network (ANN) for enhanced watermark embedding and extraction. CT enhances pixel intensity while NMF extracts the most significant features from each (m x m) block (with m= 2 in simulations), improving the method's robustness and imperceptibility. Before embedding, the watermark undergoes a new scrambling technique combining adaptive diffusion and non-linear transform, significantly enhancing security. A new embedding equation integrates the watermark into the non-negative elements of the host image blocks. Statistical features from watermarked blocks are used to create a dataset for extraction, and an ANN model trained on this dataset enables accurate watermark recovery. The proposed method achieves a PSNR of 66.43 to 77.02 dB (**31.91% improvement**), SSIM of 0.99 to 1.00 (**1.02% improvement**), and NC of 0.99 to 1.00 (**1.88% improvement**). Its computational complexity is $O(n^2)$, a **33.33% reduction** compared to traditional $O(n^3)$ methods. Security metrics include RE = 0.3094, UACI = 34.21%, and NPCR = 99.15%, demonstrating superior robustness and efficiency over existing approaches.

INDEX TERMS Image watermarking, YIQ color model, Catalan transform (CT), non-negative matrix factorization (NMF), artificial neural network (ANN), imperceptibility, and robustness.

I. INTRODUCTION

The creation, distribution, and consumption of digital content have been transformed by the accelerated advancements in internet and multimedia technologies. However, these developments have also introduced significant challenges in protecting intellectual property rights and ensuring the authenticity of digital media [2]–[4]. Digital watermarking has become an essential method for embedding identifying information inside digital output, maintaining its quality and usability. Despite extensive research in this domain, conventional watermarking techniques face several limitations, including low robustness against attacks, limited application scope, and trade-offs between imperceptibility, robustness, and watermark capacity [5], [6], [9]. A comprehensive classification of watermarking techniques is presented in Fig. 1, organized into five main categories: visibility, document type, extraction method, attack resistance, and processing domain. This classification highlights the diverse strategies used in watermarking technology, reflecting its complexity and versatility. Conventional watermarking methods, summarized in Fig. 2, include copyright markers by altering either the pixel values or transform coefficients of the host image. Although transform domain methods demonstrate greater resilience compared to spatial domain techniques, they still suffer from limitations

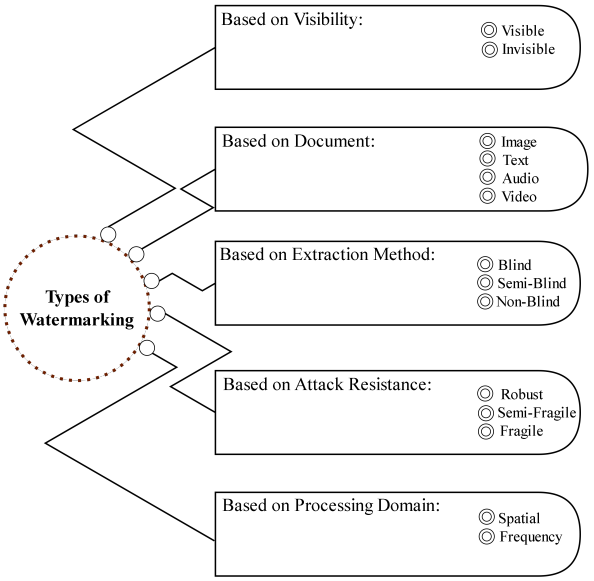


FIGURE 1: Types of watermarking techniques.

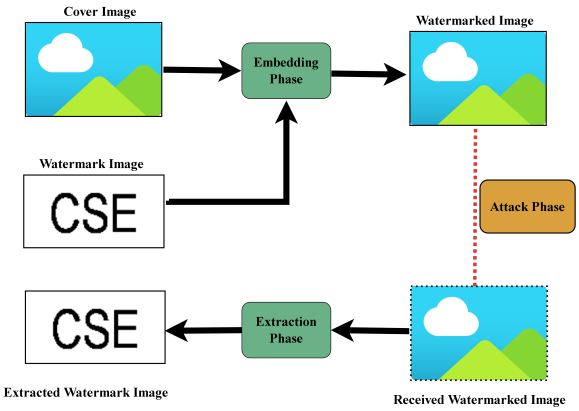


FIGURE 2: Conventional watermarking overview. It shows the embedding phase, where a watermark is inserted into a cover image, followed by a potential attack phase on the watermarked image. The extraction phase then attempts to recover the original watermark from the received watermarked image.

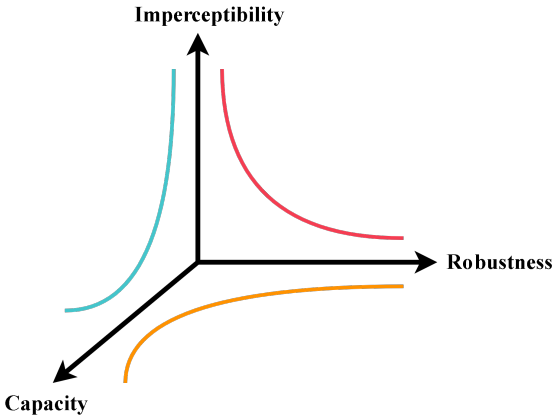


FIGURE 3: Trade-off characteristics of watermarking. The curves demonstrate that optimizing one characteristic often comes at the expense of the others, highlighting the challenge of balancing these factors in watermarking techniques.

such as lower resistance to attacks and restricted application scenarios [8]. The evaluation of watermarking techniques requires a careful analysis of three key factors: imperceptibility, robustness, and watermark capacity. Imperceptibility denotes the degree to which the watermarking procedure influences the visual quality of the source image. Robustness evaluates the watermark's capacity to endure diverse purposeful and inadvertent assaults, like compression, scaling, or noise introduction. The watermark capacity defines the maximum amount of information that can be embedded without compromising invisibility and durability. Fig. 3 illustrates the inherent trade-offs among these factors, highlighting the challenge of balancing them effectively. For example, improving robustness may increase watermark visibility, while increasing the data payload could degrade the host image quality. Deep learning-based watermarking methods have recently gained attention as a promising alternative, offering solutions to some of these challenges [7], [10]. They leverage advanced computational models to optimize watermark embedding and extraction by (a) identifying optimal embedding locations, (b) balancing invisibility and robustness through adaptive embedding strength, (c) simulating attacks to ensure extraction accuracy, and (d) reducing noise and errors during extraction. Consequently, these methods have great potential to improve watermarking performance in terms of security, adaptability, and efficiency.

To address the challenges in conventional methods, this study proposes a watermarking approach that integrates the Catalan transform (CT) and non-negative matrix factorization (NMF) for embedding with an artificial neural network (ANN) for extraction. The CT enhances pixel intensity to provide a robust foundation for embedding, ensuring better resilience to distortions. NMF is employed to extract optimum features while ensuring imperceptibility by the imposition of non-negativity restrictions, hence preserving the quality of the host image. The ANN model, structured with a multi-layer architecture, enhances the accuracy and efficacy of watermark extraction, particularly under various attack situations like compression, cropping, and noise introduction. The integration of CT, NMF, and ANN creates a synergy that balances imperceptibility, robustness, and capacity. Furthermore, the method incorporates a dynamic threshold-setting mechanism, which adapts to various conditions, significantly enhancing overall security and adaptability. This comprehensive approach aims to achieve a more effective balance between key performance metrics compared to conventional techniques. This paper is structured as follows: In Section II, a literature review is contained, which summarizes the existing literature based on our findings. In Section III, the preliminaries and

background concepts are presented. In Section IV, a thorough description of the proposed watermarking method is outlined, as well as describing the process of embedding and extraction. Section V provides a comprehensive description of the experimental setup, results, and analysis. Finally, Section VI serves as the conclusion to this paper and provides a brief summary of the major findings and potential for future development.

II. RELATED WORK

In December 1992, the term ''digital watermark'' was initially proposed by Andrew Tirkel and Charles Osborne. Gerard Rankin, Charles Osborne, Ron Van Schyndel, and Andrew Tirkel were the first individuals identified in 1993 as the creators and attackers of a broad spectrum of steganographic contiguous communication [1]. This achievement in the field was significant and was made possible by the efforts of these individuals. The practice of using watermarks began in Italy in the thirteenth century and soon spread throughout Europe. To function as unique identifiers, they were consistently created during the entire papermaking process. Originating in Italy during the thirteenth century, watermarks served as unique identifiers embedded during the manufacturing process, a practice that later spread across Europe. Digital watermarking has built upon this historical foundation to become a critical modern technology. With the advancement of the internet and multimedia technologies, the multiplication of digital material has attained unprecedented proportions. This innovation has democratized information access but has also posed considerable issues in protecting intellectual property and verifying the validity of digital material. Modern digital watermarking addresses these challenges by securely embedding unique identifiers directly within digital media.

A. CONVENTIONAL METHODS

In the context of a transform domain, watermarking is often applied as frequently modifying the transform coefficients. A range of transformations was employed during the embedding of the mark. The discrete cosine transform (DCT) is among the most often utilized procedures [10]–[13], [56]. In the paper, [16], a robust watermarking method is presented, which applies the analytical Clifford Fourier Mellin transform to embed and remove watermarks. The improved performance and security through the use of the complex Fourier transform leads to more invisible, secure systems with greater resistance to a variety of attacks. The efficacy and longevity of the method are demonstrated by numerical data with an average PSNR of 48.02 dB and an NC above 0.97. The approach is applicable to all standard image processing operations, including JPEG compression assaults. A method integrating the discrete cosine transform (DCT) with the discrete wavelet transform (DWT) was utilized in [14]. A two-level DWT was utilized to capture the high-level sub-band before segmenting it into quadrilateral segments. Each of these segments was then transformed into DCT. In the DCT block, the mid-frequency coefficients had pseudo-noise (PN) sequences representing the watermark bits. In paper [17], researchers introduced a novel technique for watermarking color document pictures. They developed a hybrid method that integrates the discrete curvelet transform (DCuT) and discrete cosine transforms (DCT) to enhance security. They assessed their outcomes based on peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), and normalized correlation (NC). They established a maximum PSNR value of 48.42 when assessing with JPEG, showing good robustness. The SSIM, NC, and UQI, respectively, presented results of 0.99, revealing excellent imperceptibility. Additionally, the method also demonstrated good tolerance to JPEG compression and several attacks, including rotation and salt & pepper noise. The approach of merging DWT and DCT was presented in the study [15]. The researchers conducted discrete wavelet transform (DWT) on the original image and subsequently on the HL1 sub-band. The HL2 sub-band was subsequently divided into non-overlapping pictures composed of four-by-four segments, and the discrete cosine transform (DCT) was applied to each block, employing the image to embed the PN-sequences of the watermark bits inside the mid-frequency coefficients of the DCT block. A method for blindly adding watermarks to images was developed in [18]. The combination of DCT, principal components analysis (PCA), and DWT was able to achieve higher PSNR and NC values while also improving robustness against attacks. A greater PSNR value signifies superior imperceptibility of the picture, indicating that the watermark is less discernible. This measure is frequently employed to evaluate the integrity of watermarked photographs. The objective of NC is to attain values around 1.00 through the comparison of original and extracted watermarks, signifying a substantial degree of resistance against assaults. In paper [19], a comprehensive solution for protecting digital content in practical applications was developed. This study introduces a method for embedding watermarks in color images via adaptive hybrid techniques inside the YCbCr color space. The adaptive embedding factor is ascertained by integrating visual and edge entropies using the discrete wavelet transform (DWT), Walsh-Hadamard transform (WHT), and singular value decomposition (SVD). The watermark is embedded in the (Y) component of the YCbCr color space and safeguarded by the Arnold transform (AT) to augment its resilience. The findings, with an average PSNR of 40.0876 dB and an average SSIM of 0.9883, unequivocally demonstrate that the method effectively attains both robustness and invisibility in watermarking. The artificial bee colony (ABC) algorithm was employed as an enhanced method for blind color picture watermarking inside the hybrid LWT-DCT domain in [20]. The analysis revealed an average PSNR of 41.130 dB, an average SSIM of 0.9937, and an average NC of 0.9957. Nonetheless, the study highlights the disadvantages and benefits of diminished PSNR values and extended computation times.

To summarise, conventional watermarking approaches face several limitations despite their varied techniques. While they often achieve good results in certain metrics, they struggle to balance imperceptibility and robustness optimally. Many are

vulnerable to specific types of attacks or image-processing operations, limiting their real-world applicability. Additionally, more complex methods involving multiple transforms or optimization algorithms can lead to increased computational overhead and longer processing times.

B. DEEP LEARNING-BASED METHODS

Deep learning has revolutionized numerous fields by providing unparalleled accuracy, robustness, and the ability to automatically extract intricate patterns from large datasets [57]–[59]. Deep learning has recently shown significant success in image processing, making it a promising option for watermarking applications [21], [22]. In paper [23], the past, current, and potential future of hybrid DWT-SVD technology in image watermarking was examined. The system employed a backpropagation neural network (BPNN) for extraction and a deep belief network (DBN) for preprocessing. Essential parameters comprise an average NC of 0.99, an average PSNR of 65.304 dB, and an average SSIM of 0.99. Nonetheless, attaining these parameters negatively affects processing durations and watermark integrity. A novel digital image watermarking method utilizing convolutional neural networks (CNNs) was presented in [24]. Traditional methods have vulnerabilities to attacks, prompting the development of deep learning-based approaches like this one. It extracts latent features from cover and secret images, combines them, and employs a denoising autoencoder network for extraction. The technique proficiently integrates watermarks, with a peak PSNR of 44.8923 dB and an SSIM score of 0.999. The NC value frequently surpassed 0.7638, demonstrating strong performance against diverse threats. A robust blind color picture watermarking technique was developed in the paper [25] utilizing the YIQ color model, the LWT methodology, and deep artificial neural networks (DANN). The mean imperceptibility of 52.48 dB in the evaluated pictures indicates that this approach effectively balances robustness and imperceptibility. Principal component analysis (PCA) is utilized to extract watermarks and get essential statistical information. The method demonstrates robustness against prevalent image assaults. In paper [26], a novel learning-based blind watermarking algorithm inspired by deep learning was proposed to enhance performance. Their approach involves enhancing performance by utilizing the cross-entropy loss function and considering watermark extraction as block classification through a modified convolutional neural network (CNN). The study emphasizes that maintaining images with good visual quality requires a PSNR exceeding 35 dB and SSIM above 0.95, metrics commonly used in image quality assessment. A watermarking system that is dependable and protected by using a cycle variable autoencoder was developed in the study [27]. The network successfully acquired the ability to encode and decode images using a 1-bit watermark, resulting in enhanced visual fidelity of the photographs. However, the technique had a low watermark capacity, which limited its practical applications. Additionally, in paper [60], the DANS model was introduced, including inception blocks and non-local sparse attention inside a U-Net architecture. This model attained a PSNR of 38.49 dB and an SSIM of 0.9622, demonstrating exceptional performance in noise-affected images. However, it faced limitations in deployment on edge devices due to processing speed constraints. Similarly, the DHTCUN architecture was proposed, combining transformers and CNNs for improved computational efficiency and feature representation in [61]. It attained a PSNR of 38.48 dB and an SSIM of 0.9624; nevertheless, the augmented parameters may influence deployment efficiency.

TABLE 1: Comparison of watermarking techniques

Method	Approach	Advantages	Observations
[14]	Hybrid DWT-DCT	Robust against compression and noise	Limited attack resistance
[16]	Analytical Clifford Fourier Mellin Transform	High imperceptibility; robust to common attacks	Computationally intensive
[19]	Adaptive Hybrid YCbCr with AT	High PSNR and SSIM values	Increased complexity; limited scalability
[23]	BPNN + DBN	High NC; imperceptibility	High computational complexity
[24]	CNN + Denoising Autoencoder	Strong robustness and imperceptibility	Limited payload
[25]	LWT + DANN + PCA	High imperceptibility; resilience to attacks	Limited payload; real-time performance
[60]	DANS Model (Inception + Sparse Attention)	Superior noise performance	Deployment challenges on edge devices
[61]	DHTCUN (Transformer + CNN)	Efficient feature representation	Increased model size impacts efficiency
Proposed	Catalan transform + NMF + ANN	High PSNR; robust to multiple attacks; low time complexity	Dependency on accurate parameter tuning

The literature review reveals several limitations in existing watermarking methods, as summarized in Table 1. Despite diverse approaches, achieving an optimal balance between imperceptibility, robustness, security, and data payload remains challenging. Many methods, such as [14], [16], show promise in specific metrics but falter in scalability or computational efficiency. The suggested technique thoroughly overcomes these issues by using the synergistic combination of Catalan transform (CT), non-negative matrix factorization (NMF), and artificial neural networks (ANN), as seen in the comparison. This technique seeks to enhance the equilibrium among essential performance measures by attaining elevated PSNR values, minimizing computing complexity, and exhibiting greater resilience to diverse assaults.

III. PRELIMINARIES

In the digital era, marked by the unparalleled accessibility of data sharing through the Internet, it is essential to utilize digital image watermarking techniques to ascertain the authenticity and ownership of digital content across various

applications, including copyright protection, content authentication, and tamper detection. The Catalan transform, an approach from combinatorial mathematics with unique properties useful in many signal processing tasks, is especially useful here, for it serves as an ideal approach to watermarking by correct usage in the applications above. Its positive and negative features are beneficial in significantly improving the imperceptibility of the watermark, as well as its robustness against known attack strategies. Further, non-negative matrix factorization (NMF) is a mathematical technique that can analyze data patterns by decomposing images into non-negative components. This property allows watermarks to be embedded and extracted from images while preserving the original image quality.

A. CATALAN TRANSFORM(CT)

In digital image processing, efficient methods for pixel transformation are essential for various applications, including compression and encryption. One such method is the Catalan transform (CT) [28], which plays a significant role in altering pixel values while preserving image integrity.

Consider a set of pixels $p_0, p_1, \ldots, p_l$ denoted as P. The CT is utilized to compute transformed components $t_0, t_1, \ldots, t_l$ using the formula:

$$t_l = \sum_{k=0}^{l} \frac{k}{2l-k} \binom{2k}{l-k} p_k \tag{1}$$

Where k represents the index of summation.

For groups of pixels with a size of 4, the transformation yields transformed quadruples as follows:

$$t_j = \begin{cases} p_0 & \text{if } j = 0 \\ p_1 & \text{if } j = 1 \\ p_1 + p_2 & \text{if } j = 2 \\ 2p_1 + 2p_2 + p_3 & \text{if } j = 3 \end{cases}$$

To revert this transformation and recalibrate pixel values $p'_0, p'_1, \ldots, p'_l$, the inverse Catalan transform (ICT) is employed. It uses the formula:

$$p'_l = \sum_{k=0}^{l} (-1)^{l-k} \frac{k}{l-k} t_k \tag{2}$$

This inverse process restores the original pixel values from the transformed quadruples, resulting in 4-pixel groups:

$$p'_j = \begin{cases} t_0 & \text{if } j = 0 \\ t_1 & \text{if } j = 1 \\ -t_1 + t_2 & \text{if } j = 2 \\ -2t_2 + t_3 & \text{if } j = 3 \end{cases}$$

If no modifications are done, the recalculated pixel values will remain exactly the same as the original values, ensuring that $p'_j = p_j$.

B. NON-NEGATIVE MATRIX FACTORIZATION (NMF)

Non-negative matrix factorization (NMF) is a widely used technique for dividing non-negative matrices into two components: W, which contains basis vectors, and H, which contains coefficients [36]. This factorization captures fundamental data structures and ensures interpretability. The objective of NMF is to identify two non-negative matrices W and H that, when compared to a non-negative matrix V, have the same shape as $V \approx WH$. The coefficients are contained in H, while the basis vectors are included in W. The mathematical representation of NMF is shown as follows:

$$\min_{W,H} \|V - WH\|_F^2 \tag{3}$$

Subject to:

$$W \geq 0$$
$$H \geq 0$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

Using techniques such as alternating least squares or multiplicative updates, the algorithm iteratively updates W and H until convergence. The fundamental patterns and coefficients are represented by the factorized matrices W and H, respectively. The original matrix V is reconstructed by multiplying W and H together. The objective of this procedure is to extract meaningful features from the original data, reduce dimensionality, and preserve essential information.

C. ADAPTIVE DIFFUSION AND NON-LINEAR TRANSFORM-BASED SCRAMBLING

The combination of adaptive diffusion [28] and non-linear transform [29] provides a robust method for scrambling and subsequently descrambling data. In this technique, each element in the data array is modified by adding noise and then applying a non-linear transformation. For each element i in the original data array represented by $P_o(i)$, the scrambling process involves the following steps:

1) **Adding Noise:** The element $P_s(i)$ in the scrambled array is obtained by adding a noise component $N(i)$ to the original element:
$$P_s(i) = P_o(i) + N(i) \tag{4}$$
2) **Non-linear Transformation:** The modified element $P_s(i)$ is then raised to the power of α, where α is the strength parameter for the non-linear transformation:
$$P_s(i) = (P_s(i))^{\alpha} \tag{5}$$

The descrambling procedure is implemented to retrieve the original data array from the scrambled version. This involves the following steps for each element i in the scrambled array represented by $P_s(i)$:

1) **Inverse Non-linear Transformation:** The element $P_d(i)$ in the descrambled array is obtained by taking the $\frac{1}{\alpha}$ th power of the scrambled element:
$$P_d(i) = (P_s(i))^{\frac{1}{\alpha}} \tag{6}$$

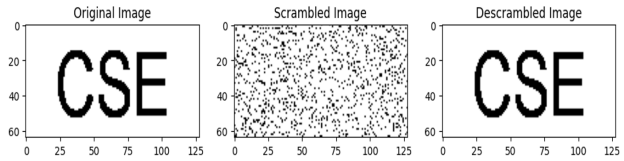


FIGURE 4: Demonstration of image scrambling and descrambling. Left: Original image, Centre: Scrambled image, Right: Descrambled image.

2) **Removing Noise:** The noise component $N(i)$ is subtracted from the descrambled element to obtain the final value:

$$P_d(i) = P_d(i) - N(i) \tag{7}$$

- **Adaptive Diffusion Strength:**

The strength of the adaptive diffusion process, which is essential for image processing methods like the one shown in Fig. 4, is established through a function evaluating local variance among the data. This computational process employs a Gaussian filter characterized by a predetermined window size. The fundamental parameter, denoted as the base strength, establishes the overarching scaling factor for the diffusion strength. Concurrently, the window size parameter governs the scope of the local neighborhood under consideration for variance computation. The formula for adaptive diffusion strength is:

$$\text{Strength} = \text{Base Strength} \times \text{Local Variance} \tag{8}$$

D. STATISTICAL FEATURES FOR IMAGE CHARACTERIZATION

Statistical features represent quantitative measures that characterize the properties of a dataset or image. They can provide important insights into a dataset's distribution, central tendency, dispersion, and other characteristics. Researchers and analysts can identify, evaluate, and analyze statistical features to understand spatial patterns, trends, and relationships in data. Below are extended definitions of each of the statistical features:

1) **Entropy**: Entropy is a measurement of uncertainty or unpredictability in the data of images [31]. It assesses whether the distribution of pixel intensities in the image contains information or is random. If the Entropy is low, the pixel intensities are predictable and distributed equally; in the case of high Entropy, the pixel intensities are distributed randomly or inconsistently. Entropy has been used in various fields, including image compression, texture analysis, and image segmentation, as a measurement of an image's richness or complexity information.
2) **Mean**: The center value of the picture pixel intensities is represented by the Mean, which is also called the arithmetic average [32]. To obtain the final result, it is necessary to divide the whole sum of pixel values by the total number of pixels. To determine the brightness or intensity of a photograph, determining its average value is a suitable initial step. Image enhancement, normalization, and background reduction are just a few examples of how this basic statistical metric is used in image processing.
3) **Mode**: Mode refers to the value or category that appears in the image data the most frequently. The pixel intensity value that transpires most frequently throughout an entire image is ascertained [32].
4) **Median**: The Median represents the central value of a set of arranged picture pixel values. Not only does it reliably indicate the central tendency, but it also separates the top half of the intensity range of the pixels from the bottom half.
5) **Quartiles**: The Median is situated within the second quartile, which is obtained by dividing the list of picture pixel values into four equal halves. Based on the statistics, 25% of the pixel values are lower than the first quartile (Q1), and 75% of the pixel values are lower than the third quartile (Q3).
6) **Moment**: Moments are quantitative measures used to describe the shape of a probability distribution [33]. In images, they characterize the distribution of pixel values, including central tendency, spread, symmetry, and peakedness.
7) **Variance**: Variance quantifies the extent to which the pixel values of a picture deviate from their average value, indicating the level of dispersion or spread. A larger Variance signifies that the pixel values exhibit a greater degree of dispersion from the Mean.
8) **Covariance**: Covariance quantifies the relationship or dependence between the intensities of different pixels or image regions [34]. It measures the joint variability of two random variables, such as pixel values.
9) **Standard Deviation**: By calculating the standard deviation, which is the square root of the Variance, one may have a better understanding of the pixel value distribution of a picture.
10) **Kurtosis**: When comparing a pixel value distribution to a normal distribution, Kurtosis indicates how flat or peaky the distribution is. Outliers, or very high or low numbers, in the picture data are detected.
11) **Skewness**: Skewness measures the asymmetry or lack of symmetry in the distribution of pixel intensities around the Mean value [35]. It quantifies the degree of asymmetry in the image data.
12) **Coff_1, Coff_2, Coff_3, and Coff_4**: Frequently, these are coefficients or features derived from the image data that are peculiar to the particular technique being presented.

IV. METHODOLOGY

This section introduces a new methodology for digital image watermarking, highlighting numerous significant breakthroughs that tackle essential difficulties in the domain. The proposed method aims to enhance the imperceptibility, ro-

bustness, security, and efficiency of watermarking systems while maintaining a high data payload. The proposed methodology introduces several key innovations in digital image watermarking:

1) Development of a new watermarking technique combining Catalan transform and non-negative matrix factorization, previously unexplored in this combination, to enhance imperceptibility and robustness in digital image watermarking.
2) Introduction of an adaptive threshold calculation method coupled with a new embedding equation designed to optimize overall watermarking performance.
3) Creation of a sophisticated extraction procedure that harnesses the power of deep neural networks and principal component analysis, ushering in a new paradigm for accurate watermark recovery.
4) Development of a new scrambling technique termed ''adaptive diffusion and non-linear tansform-based scrambling'' to enhance watermark security.
5) Attainment of marked improvements across multiple performance metrics, including PSNR, SSIM, NC, BER, RE, UACI, and NPCR, alongside enhanced data payload and reduced time complexity.

These contributions all add up to providing a watermarking system that delivers improved performance in imperceptibility, robustness, extraction accuracy, security, data payload, and computational efficiency. The added security from the new scrambling technique provides additional protection against unauthorized access or tampering. The same case applies to the improved data payload that will support more flexible and comprehensive watermarking applications and lower the enumerated time complexity. Embedded watermarks make this system more practical for real-time or large-scale implementations. The suggested approach consists of two main procedures: watermark embedding and watermark extraction. The watermark embedding process begins with converting a cover image into the YIQ color space, followed by applying the Catalan transform and non-negative matrix factorization. Afterwards, the adaptive threshold calculation method is used to determine the optimal embedding strengths, High imperceptibility can be achieved. Moreover, a new scrambling technique will scramble the watermark before embedding it, improving security. On the other hand, in the extraction process, deep neural networks are used with the aid of principal component analysis to recover the embedded watermark accurately. Therefore, it efficiently recovers the watermark under various manipulations/attacks. Ensuring efficiency in the proposed embedding and extraction methods: It reduces the complexity of the time associated with the process without decreasing the watermarking quality. Thus, it allows the proposed system to be applied for real-time processing or large volumes of image handling. This method advances watermarking, requiring high efficiency. The method of watermarking, from its embedding to its extraction, is explained in detail in Fig. 5 and Fig. 6, respectively.

A. EMBEDDING PROCEDURE

Fig. 5 presents a comprehensive schematic representation of the proposed embedding procedure. This diagram serves as a visual roadmap, elucidating each stage of the embedding algorithm with precision and clarity. The embedding procedure begins by obtaining a color cover image and a watermark image that adheres to predetermined dimensional standards. It progresses through a series of transformations and integrations, each represented by a distinct component in the diagram. The algorithm decomposes the cover image into its YIQ components, with subsequent operations focusing on the ''I'' component. This component undergoes partitioning into (2x2) blocks, each subjected to the Catalan transform. The resulting coefficients are then shuffled using secret keys and applying non-negative matrix factorization (NMF).

A critical process phase entails manipulating coefficient bits by mathematical formulas related to the watermark bits. The last stages involve reassembling the components and reordering the seed keys via the inverse Catalan transform. The process culminates in creating the watermarked image, which maintains the precise dimensions of the original cover image. The written explanation is complemented by the algorithmic representation, which enhances understanding of the complex embedding process. Presenting a simple and understandable explanation of the proposed methodology allows readers to understand the complex algorithmic subtleties of watermarking more deeply.

Algorithm 1 Threshold Calculation

Require: Original Image to be Watermarked
Ensure: Threshold value
Histogram calculation:
Calculate histogram $X = \{x_i\}, i = 0, 1, \ldots, 255$
where $x_i = \frac{n_i}{N}$, n_i is the count of pixels with intensity i
and N is the total number of pixels
CDF calculation:
Compute $CDF = \{C_i\}, i = 0, 1, \ldots, 255$
where $C_i = \sum_{j=0}^{i} x_j$
Random value generation:
Generate random value $R \sim U(0, 1)$
Find corresponding bin:
Identify bin $k = \min\{i : C_i \geq R\}$
Map random value to pixel intensity:
$V = \text{Interpolate}(R, [C_{k-1}, C_k], [b_{k-1}, b_k])$
Normalization and return: Threshold$= V/255$

As delineated in Algorithm 1, the threshold calculation technique is a sophisticated and crucial component in image processing and watermarking approaches, designed to determine an optimal threshold value adaptively. The process initiates with a comprehensive histogram calculation, where the frequency distribution of pixel intensities is meticulously computed and normalized, resulting in a set $X = x_i$ for intensities i ranging from 0 to 255. This is followed by the cumulative distribution function (CDF) derivation, represented

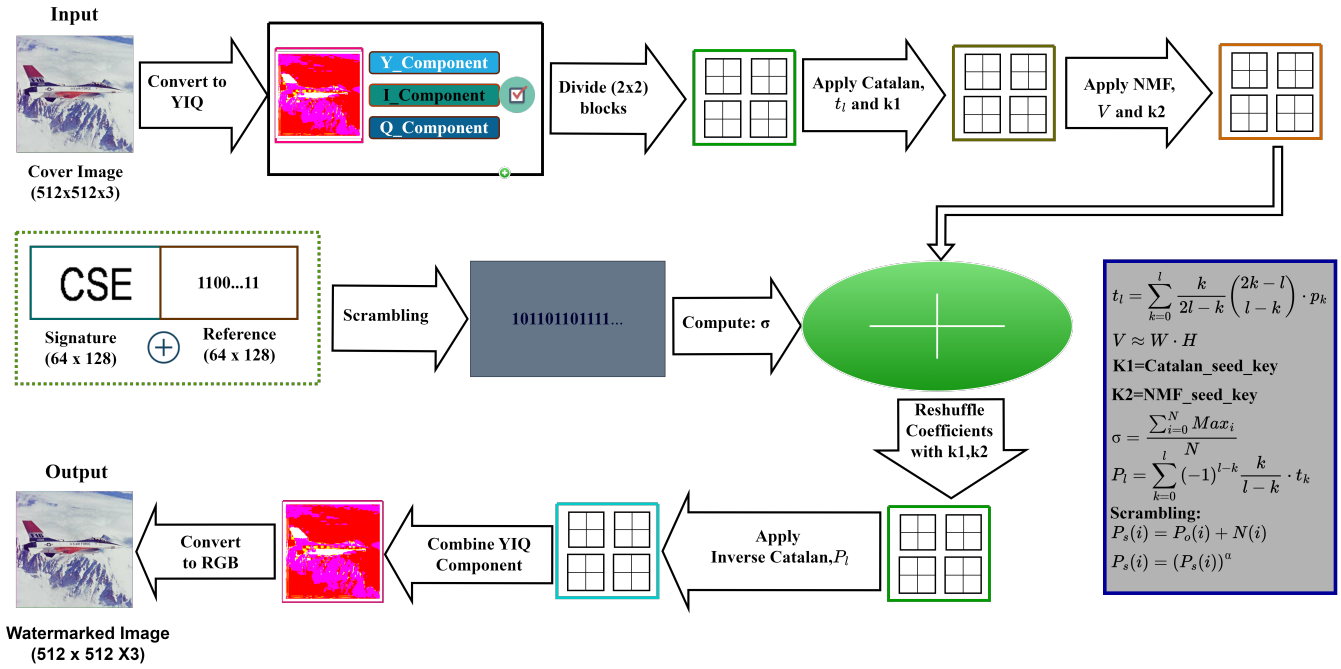


FIGURE 5: Schematic representation of the embedding process, illustrating the sequential stages from initial image acquisition to final watermark integration.

as $CDF = C_i$, which accumulates the normalized frequencies up to each intensity level. To introduce stochasticity and enhance robustness, the algorithm generates a random value R from a uniform distribution $U(0,1)$. This random value is then used to identify a corresponding intensity bin k in the CDF, where k is the minimum value satisfying $C_k \geq R$. To refine the threshold and account for the discrete nature of intensity levels, the algorithm employs interpolation between the identified bin and its predecessor, mapping the random value R to a pixel intensity V. This interpolation step represented as $V = \text{Interpolate}(R, [C_{k-1}, C_k], [b_{k-1}, b_k])$, where b represents bin edges, significantly enhances the accuracy of the threshold determination. Finally, the interpolated intensity value V is normalized by dividing it by 255, yielding a consistent threshold value across different images. This iterative process of random value generation, bin identification, interpolation, and normalization ensures that the resulting threshold is adaptive to the specific image characteristics and includes an element of unpredictability, making it particularly effective for various image processing and watermarking applications where both precision and security are paramount.

1) Embedding Equations

This subsection provides a detailed description of the embedding equations, which are fundamental to embedding watermark bits in the matrix $\mathbf{W}$. These equations adaptively adjust matrix coefficients to ensure imperceptibility while maintaining robustness against distortions. The embedding process modifies $\mathbf{W}$ based on the watermark bit values and the differences in matrix coefficients.

1) **Condition 1: For watermark_bit $= 1$ and $\Delta_{\max}[i] \geq \max(\sigma, \tau)$:** The largest coefficient in the block is updated as:

$$W[i][j_{\max}] = \max(0, V_{\max} - \tau), \tag{9}$$

where:

- $W[i][j_{\max}]$ is the largest coefficient in the block i.
- $V_{\max}$ represents the value of the largest coefficient.
- τ is a dynamically computed threshold.
- $\Delta_{\max}[i]$ is the difference between the largest and second-largest coefficients in block i.

2) **Condition 2: For watermark_bit $= 1$ and $\Delta_{\max}[i] < \max(\sigma, \tau)$:** No modification is applied to $\mathbf{W}$ to preserve insignificant coefficient differences.

3) **Condition 3: For watermark_bit $= 0$:** The smallest coefficient in the block is updated to:

$$W[i][j_{\min}] = V_{\max} - \Delta_{\max}[i], \tag{10}$$

where:

- $W[i][j_{\min}]$ is the smallest coefficient in the block i.
- $j_{\min}$ is the index of the smallest coefficient.

The parameters σ and τ are critical to dynamically adjusting the embedding process:

- τ is derived using the adaptive threshold calculation outlined in Algorithm 1.
- σ represents the Mean of the largest coefficients in all blocks, calculated as:

$$\sigma = \frac{1}{N} \sum_{i=1}^{N} V_{\max,i} \tag{11}$$

where N is the total number of blocks and $V_{\max,i}$ is the largest coefficient in block i.

Equation (11) ensures that the threshold adapts to the global characteristics of the image, providing a balance between embedding strength and imperceptibility. By averaging the maximum coefficients across all blocks, σ captures the overall intensity distribution, ensuring consistent watermark embedding in varying image regions. These embedding equations enable dynamic and localized adjustments to $\mathbf{W}$, ensuring robustness and imperceptibility. The use of $\Delta_{\max}[i]$, σ, and τ allows the process to adapt to both global and local image characteristics, improving the resilience of the watermark against attacks such as compression or noise addition.

2) Watermark Embedding Algorithm

Watermark embedding is a process where, with much security and imperceptibility, a watermark is allowed into the cover image. This process is summarized in Algorithm 2, which describes the steps to be carried out sequentially. The embedding steps are described below, as taken from the algorithm.

I. Preprocessing (Steps 1–2 in Algorithm 2): The cover image is transformed into the YIQ color space, and the "I" component of the YIQ is selected for watermark embedding. This "I" component is split into blocks of non-overlapping (2×2) to enable localized watermark insertion.

II. Transformation and shuffling (Steps 3–5): For the robustness of watermark embedding, each block with (2×2) is processed with the Catalan transform to expand the feature space. The output coefficients are randomized according to the secret key K_1 to protect the embedding process. These blocks are then split by a non-negative matrix factorization (NMF). Again, the coefficients are shuffled using the second secret key K_2 to provide more security.

III. Threshold and scrambling (Steps 6–7): The average coefficient difference across the blocks is calculated by Equation 11 to calculate the dynamic threshold for embedding. The signature and reference watermarks are integrated and scrambled for their security improvement issues before embedding.

IV. Coefficient modification (Steps 8–13): The difference between the two largest coefficients is calculated for each coefficient bit in the blocks:

- If the watermark bit is 1, the largest coefficient is modified using Equation 9.
- If the watermark bit is 0, the smallest coefficient is modified using Equation 10.

This step ensures that the watermark is embedded robustly and imperceptibly by selectively altering the coefficients.

V. Final adjustments and reconstruction (Steps 14–16): The altered blocks are rearranged utilizing the keys K_1 and K_2, and the inverse Catalan transform is employed to recreate the "I" component. The "Y", "I", and "Q" components are ultimately recombined to create the watermarked image, maintaining the proportions and visual integrity of the original cover image.

Algorithm 2 Watermark Embedding Algorithm

Require: Cover Image $(512 \times 512 \times 3)$, Watermark Image (64×128), Keys K_1, K_2
Ensure: Watermarked Image $(512 \times 512 \times 3)$

```
Apply YIQ color model to cover image; extract "I" component
Split "I" component into (2 × 2) blocks
Apply Catalan transform to each block
Shuffle coefficients using key K1
Apply NMF and shuffle coefficients using key K2
Determine avg. coefficient difference using eq. 11
Integrate signature and reference watermark; then apply scrambling
for each coefficient bit do
    Calculate difference between two largest coefficients
    if Watermark bit = 1 then
        Modify largest coefficient using eq. 9
    else
        Modify smallest coefficient using eq. 10
    end if
end for
Reshuffle blocks and coefficients using K1, K2
Apply inverse Catalan transform
Recombine "Y", "I", "Q" components for watermarked image
```

B. ANN-BASED EXTRACTION PROCEDURE

The watermark extraction procedure guarantees precise retrieval of the embedded watermark after its existence has been confirmed. This technique is elucidated in Algorithm 3, which delineates the sequential procedure. The principal phases of the extraction process are outlined here, aligning with the steps in the algorithm.

I. Preprocessing (Steps 1–2 in Algorithm 3): The watermarked picture is transformed into the YIQ color space to extract the "I" component for subsequent processing. The "I" component is partitioned into non-overlapping (2×2) units to facilitate localized feature extraction.

II. Transformation and shuffling (Steps 3–5): Every block is put through the Catalan transform, which increases the feature space for the purpose of detecting watermarks that are resistant to damage. To ensure the confidentiality of the information, the revised coefficients are permuted with the use of the secret key $K1$. Following that, non-negative matrix factorization, also known as NMF, is utilized in order to partition the blocks into substantial components. The coefficients are further shuffled using a supplementary secret key, denoted by the letter $K2$, in order to increase the level of security.

III. Feature extraction and dimensionality reduction (Steps 7–8 in Algorithm 3): Divide the 16,384 (2×2) blocks into two groups: training blocks (block 8192 to block 16383) and testing blocks (block 0 to block 8191). The NMF-decomposed matrices generate a feature set comprising statistical characteristics such as Mean, Mode, Median, Quartiles,

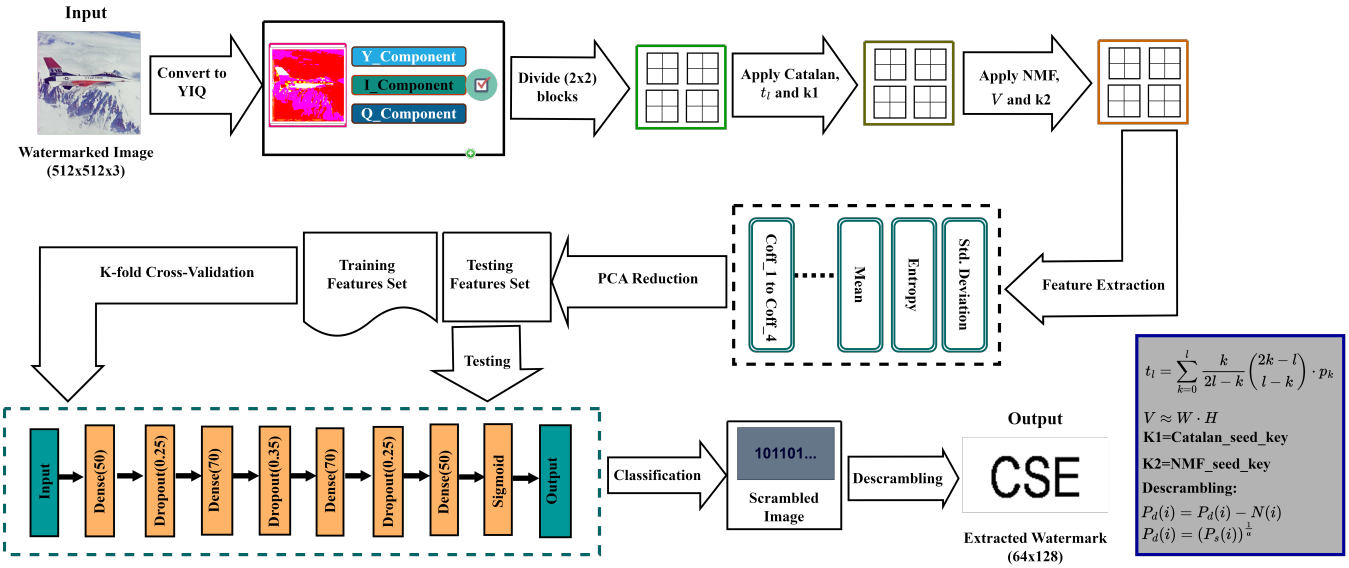


FIGURE 6: Schematic representation of the watermark extraction process. This figure illustrates the sequence of operations from initial image processing to final watermark recovery, including color space conversion, block division, transformation, randomization, dimensionality reduction, and neural network-based classification.

Moment, Variance, Covariance, Standard Deviation, Kurtosis, Skewness, and (Coff_1, Coff_2, Coff_3, Coff_4), for each block. When it comes to differentiating watermarked regions from non-watermarked ones, these characteristics are very necessary. Principal component analysis (PCA) is carried out in order to keep the ten characteristics that are considered to be the most important while also reducing the dimensionality of the features. By maintaining information that is computationally relevant, this phase contributes to the complete elimination of redundancy.

IV. Watermark detection using neural networks (Steps 9–10 in Algorithm 3): Following the reduction of the feature set, the datasets for training and testing are separated. The training set is comprised of features from reference watermark blocks, which range from block 8192 to block 16383. On the other hand, the testing set is comprised of features from the original watermark blocks, which range from block 0 to block 8191. The training data is used to train a feedforward neural network, and five-fold cross-validation is utilized in order to generalize the model and prevent it from being overfit. On the basis of the PCA-reduced features, the trained neural network makes a prediction regarding whether each block has a watermark bit, which may be either a "1" or a "0".

V. Watermark reconstruction (Steps 11–12 in Algorithm 3): The neural network outputs of each block are concatenated to reconstruct the watermark sequence. Since the watermark was scrambled during the embedding phase for security, a descrambling process, the inverse of the original permutation, is applied using the secret keys. The descrambled sequence is reshaped into its original dimensions (64×128) binary bits).

Algorithm 3 Watermark Extraction Algorithm using ANN

Require: Watermarked Image ($512 \times 512 \times 3$), $K1$, $K2$
Ensure: Extracted Watermark Image (64×128)

1: Apply YIQ color model on the watermarked image and extract the "I" component
2: Split the watermarked image into (2×2) blocks
3: Apply the Catalan transform to each block
4: Shuffle the obtained coefficients using $K1$
5: Apply NMF and shuffle the coefficients using $K2$
6: Split the 16384, (2×2) blocks into two groups, each containing 8192 for training and testing
7: Generate the feature set $\{f_{\text{train}}\} = \{1, 2, \ldots, 15\}$ of blocks containing embedded reference watermark information
8: Use principal component analysis (PCA) to obtain a set of 10 reduced features
9: Train the deep neural network using the training set and five-fold cross-validation to obtain a trained model
10: Form the testing set $\{f_{\text{test}}\}$ using blocks containing embedded signature watermark bits
11: Extract the watermark by testing the model with the testing set
12: Apply descrambling and reshape the extracted watermark to size (64×128)

1) Extraction Model Description

The watermark extraction model utilizes a multi-layer feedforward neural network, as seen in Fig. 7. This architecture features fully connected layers interspersed with dropout layers and parametric rectified linear unit (PReLU) activation functions to mitigate overfitting and capture complex nonlinearities in the data. PReLU activation functions are applied

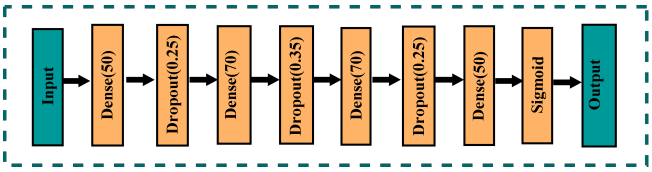


FIGURE 7: Schematic of the watermark extraction model with fully connected layers, dropout layers, and PReLU activation functions.

after each linear transformation within the hidden layers, enhancing the network's ability to model intricate relationships in the data. Dropout layers are included after the first, second, and third hidden layers with dropout rates of 0.25, 0.35, and 0.25, respectively, to improve generalization and reduce overfitting. The output layer contains a single neuron activated by a sigmoid function, which produces a probability score indicating the likelihood of the watermark's presence in the input image. This model is designed to effectively learn and extract watermark patterns from images, ensuring accurate detection and extraction.

C. RULE-BASED (CONVENTIONAL) EXTRACTION PROCEDURE

Algorithm 4 Rule-Based Watermark Extraction Algorithm

Require: Watermarked Image ($512 \times 512 \times 3$), Secret Keys (K_1, K_2)
Ensure: Extracted Watermark (64×128)
1: Convert watermarked image to YIQ color space
2: Extract "I" component
3: Divide "I" into non-overlapping 2×2 blocks
4: Apply Catalan transform to each block
5: Shuffle coefficients using key K_1
6: Apply non-negative matrix factorization (NMF)
7: Shuffle NMF coefficients using key K_2
8: Extract mean and standard deviation features
9: Normalize features: norm_feat $\leftarrow \frac{\text{feat}-\mu(\text{feat})}{\sigma(\text{feat})}$
10: Initialize pred_pixel $\leftarrow [\,]$
11: **for all** feature in norm_feat **do**
12: Compute score: score $\leftarrow \mu(\text{feat}) + \sigma(\text{feat})$
13: Compute threshold: th $\leftarrow \mu(\text{norm_feat}) + (0.5 \times \sigma(\text{norm_feat}))$
14: $y_{\text{pred}} \leftarrow \begin{cases} 1 & \text{if score} > \text{th} \\ 0 & \text{otherwise} \end{cases}$
15: pred_pixel $\leftarrow$ pred_pixel $\cup \begin{cases} 0 & \text{if } y_{\text{pred}} = 0 \\ 1 & \text{otherwise} \end{cases}$
16: **end for**
17: Apply descrambling and reshape extracted watermark to (64×128)

Algorithm 4 presents a comprehensive approach for extracting watermarks using rule-based methodologies. The procedure begins with a watermarked color image of dimensions ($512 \times 512 \times 3$) and requires two secret keys (K_1 and K_2) for the extraction process. The algorithm outputs an extracted watermark of size (64×128). The extraction process comprises several key steps. Initially, the watermarked image undergoes color space conversion from RGB to YIQ, focusing specifically on the "I" component for further processing. This component is then subdivided into non-overlapping blocks of size (2×2). The Catalan transform is applied to these blocks to capture the inherent spatial relationships within the image. To enhance security, the algorithm employs a two-stage shuffling mechanism using the provided secret keys. The first shuffling operation, performed using key K_1, is applied to the transformed coefficients. Subsequently, non-negative matrix factorization (NMF) is employed to decompose the shuffled data, followed by a second shuffling operation using key K_2.

The feature extraction phase involves computing statistical measures, specifically the mean and standard deviation. These features undergo normalization using:

$$\text{norm_feat} = \frac{\text{feat} - \mu(\text{feat})}{\sigma(\text{feat})} \tag{12}$$

For each normalized feature, the algorithm computes a score and threshold value. The score is calculated as the sum of the feature's mean and standard deviation, while the threshold is determined by adding half the standard deviation to the mean of the normalized features. The binary watermark bits are predicted based on the comparison between these scores and thresholds, with pixel values assigned as either 0 or 1.

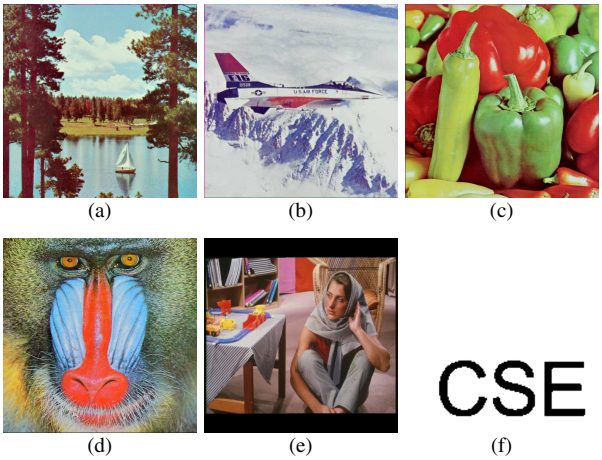
Finally, the extracted watermark undergoes descrambling and is reshaped to the original dimensions of (64×128). This rule-based approach ensures robust watermark extraction while maintaining the security of the embedded information through the dual-key encryption mechanism.

V. RESULTS AND DISCUSSION

This section assesses the watermarking approach utilizing several test photos, color models, and metrics (PSNR, SSIM, NC, BER). We further examine imperceptibility, robustness, security, data payload, temporal complexity, and trade-offs.

A. IMAGES USED FOR SIMULATION

Various RGB-format images were employed in this simulation, including well-known photographs such as Sailboat, Jetplane, Pepper, Mandrill, and Barbara in Fig. 8. Each image was standardized to dimensions of (512×512) pixels. These images were sourced from the USC-SIPI [37] and CVG-UGR [38] databases.



(a) (b) (c) (d) (e) (f)

FIGURE 8: Host images: (a) Sailboat, (b) Jetplane, (c) Pepper, (d) Mandrill, (e) Barbara. (f) Watermark (64x128 pixels).

Due to their distinctive features and properties, these images are referred to as benchmark images for the field of computer vision and image processing tasks. The embedding and extraction procedures were tested across these images to ensure their invisibility and robustness. This is indeed a comprehensive evaluation because the broad span of visual content, textures, and complexities is taken into consideration in this set of images. For instance, the Sailboat image includes smooth regions along with detailed textures, while the Mandrill image presents high-frequency textures. The performance of the algorithm is further tested on sharp edges and uniform areas by the Jetplane image, while the Pepper image provides a variety of color intensities along with curved surfaces.

B. EVALUATION CRITERIA

1) Peak Signal-to-Noise Ratio (PSNR)

PSNR is a metric commonly used to assess the quality of images. It is determined by contrasting the watermarked image with the original cover image. Mathematically, this could be stated as:

$$\text{PSNR} = 20 \cdot \log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right) \tag{13}$$

There are two important factors for calculating PSNR: MAX and MSE. MAX represents the maximum possible pixel value of the image (e.g., 255 for an 8-bit image), while MSE (Mean Squared Error) quantifies the average squared difference between the original cover image and the watermarked image. The MSE is calculated using the equation:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(I(i) - K(i)\right)^2 \tag{14}$$

In this equation, N represents the total number of pixels, I(i) denotes the pixel value of the original image, and K(i) denotes the corresponding pixel value of the watermarked image. The summation is performed over all pixels to obtain the average squared difference.

2) Structural Similarity Index Measure (SSIM)

To find out which one is superior, SSIM compares the original cover with the watermarked version based on brightness, intensity difference, and layout. The returned value falls within the range of -1 to 1, with 1 denoting an impeccable likeness. The mathematical formula for SSIM follows:

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{15}$$

Where:

- x and y are the input images (original and watermarked).
- μ_x and μ_y are the means of x and y respectively.
- σ_x^2 and σ_y^2 are the variances of x and y respectively.
- σ_{xy} is the covariance of x and y.
- C_1 and C_2 are small constants to prevent instability.

The SSIM calculation involves several components related to the input images x and y (original and watermarked). These include the means (μ_x and μ_y) of x and y respectively, the variances (σ_x^2 and σ_y^2) of x and y respectively, and the covariance (σ_{xy}) between x and y. Additionally, small constants C_1 and C_2 are included to prevent instability in the calculation, particularly when dealing with regions of near-zero mean or variance. These constitute the various components of the SSIM formula to give an overall measure of structural similarity between the original and watermarked images.

3) Normalized Correlation (NC)

NC quantifies the closeness or similarity between the extracted and original watermark images. A correlation coefficient of 1 indicates that the two signals are very closely related. The mathematical computation for the NC is given below:

$$\text{NC}(x,y) = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{16}$$

Here, x represents the extracted watermark, y represents the original watermark, $\bar{x}$ and $\bar{y}$ denote the means of x and y respectively, and N is the total number of elements in the signals. These assessment measures provide detailed information about the performance of our watermarking technique, allowing us to test its robustness, accuracy, and efficiency in real-world circumstances. Through thorough experimentation and analysis, we hope to demonstrate the usefulness of our methodology in maintaining the security and authenticity of digital photographs in various applications.

4) Bit Error Rate (BER)

Bit error rate (BER) is a crucial metric in digital watermarking that quantifies the accuracy of the extracted watermark compared to the original. It measures the proportion of bits that have been altered during the watermarking process and subsequent extraction. In watermarking literature, BER is typically calculated using the following formula:

$$BER = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} X(i,j) \oplus X'(i,j) \tag{17}$$

where:

- $X(i,j)$ is the original watermark bit at position (i,j)
- $X'(i,j)$ is the extracted watermark bit at position (i,j)
- m and n are the dimensions of the watermark
- $\oplus$ denotes the XOR operation

The BER calculation involves comparing the original watermark bits $X(i,j)$ with the extracted watermark bits $X'(i,j)$ at each position (i,j). The dimensions of the watermark are denoted by m and n. The XOR operation, represented by $\oplus$, is used to determine bit differences between the original and extracted watermarks. These components are combined in the BER formula to quantify the proportion of bit errors in the extracted watermark compared to the original, providing a measure of the watermarking system's robustness and accuracy in preserving the embedded information.

C. EXPERIMENTAL SETUP

To assess the performance of the proposed watermarking approach, a set of experiments with standard test images has been carried out. These images were chosen for hosting, from smooth to very complicated patterns, which helps to understand the strength of the proposed algorithm under extreme conditions. Each host image was resized to (512×512) pixels. The size of the watermark image was chosen as (64×128) pixels.

TABLE 2: Parameters for simulation.

Parameter	Value
Image size	512×512 pixels
Watermark size	64×128 pixels
Color model	YIQ
Block size	2×2
Transformation	Catalan transform
Factorization	Non-negative matrix factorization
Scrambling	Adaptive diffusion and non-linear transform
Neural network	Multi-layer feedforward

The experiments used the YIQ color model to split the luminance and chrominance components for efficient embedding. The Catalan transform was done to enhance the distribution of pixel intensity, while the non-negative matrix factorization ensured effective decomposition of blocks in an image. To add security, the watermark was scrambled using adaptive diffusion and non-linear transform techniques before embedding. For the extraction of the watermark, a multilayer feedforward neural network was used to recover the embedded watermark with high accuracy even under various distortions. The network was trained and tested using blocks that contained statistical features derived from the watermarked images. Table 2 summarizes the key parameters used in the experiments.

D. EXPERIMENTAL RESULTS

This subsection presents a detailed analysis of our experimental results, which were conducted using various color models to assess the robustness and efficiency of the proposed watermarking technique. The experiments were designed with great care to analyze the performance of watermark embedding and extraction procedures on a broad set of standardized images, providing an in-depth look at the method's capabilities under various conditions. We carried out two main experiments: one using the most common RGB color space and another using the YIQ color space, where luminance and chrominance information are separated, for watermark processing. We tested each color model with a suite of standard test images that are in everyday use for research involving image processing, including but not limited to-"Sailboat," "Jetplane," "Pepper," "Mandrill," and "Barbara." These images were selected because of their wide range of textures, color distributions, and complexities to ensure that our method was put through its paces. These images, with varied textures, colors, and complexities, thoroughly evaluate our method.

1) Comparison of ANN-Based and Rule-Based (Conventional) Extraction Methods

TABLE 3: Comparative analysis of ANN-based and Rule-based watermark extraction methods under various attacks.

Attack type	ANN-based extraction (NC)	Rule-based extraction (NC)
No Attack	**1.0000**	0.9748
Speckle Noise (0.01)	**1.0000**	0.8852
Sharpening (0.01)	**1.0000**	0.9021
Rotation (10°)	**0.7881**	0.6521
Salt & Pepper Noise (0.01)	**0.9914**	0.9154
Average Filtering (3×3)	**0.5301**	0.4811
Median Filtering (3×3)	**0.9057**	0.7857
Gaussian Noise	**0.9170**	0.7270
Histogram Equalization (0.5)	**1.0000**	0.9150
Brightness Adjustment (1.5)	**1.0000**	0.9250
JPEG (QF=90)	**0.9580**	0.8280
Cropping (10%)	**0.5377**	0.4307

Table 3 presents a comprehensive comparison between ANN-based and rule-based watermark extraction methods under various attack scenarios. The performance is evaluated using the normalized correlation (NC) metric, where values closer to 1 indicate better extraction quality. The ANN-based approach demonstrates superior robustness across most attack types, achieving perfect extraction (NC = 1.0000) under no attack, speckle noise, sharpening, histogram equalization, and brightness adjustment conditions. Both methods show vulnerability to geometric attacks such as rotation and cropping, with NC values dropping to 0.7881 and 0.6521 for rotation, and 0.5377 and 0.4307 for cropping, respectively. The ANN-based method maintains consistently higher NC values across all attack scenarios, with notable improvements in resilience against noise attacks (speckle, Gaussian, and salt & pepper) and filtering operations. The performance gap is particularly

evident in challenging scenarios such as Gaussian noise, where the ANN-based method achieves an NC of 0.9170 compared to 0.7270 for the rule-based approach. These results indicate that the learning-based nature of the ANN approach provides enhanced adaptability and robustness against various image processing operations and attacks compared to conventional rule-based extraction methods.

2) Experiment with the RGB Color Model

Fig. 9, 10, and 11 exhibit the outcomes of our tests with the red, green, and blue channels, respectively. Each figure showcases a comprehensive (5x4) grid of images, illustrating the watermarking process for five widely-used standard test images: Sailboat, Jetplane, Pepper, Mandrill, and Barbara. These images were chosen for their diverse characteristics, ranging from smooth textures to complex patterns, ensuring a thorough evaluation of our watermarking technique.

FIGURE 9: Red channel watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

Fig. 9 provides a detailed illustration of the results for the red channel. For each of the five test images, the figure presents four key components: the original image (column 1), the watermark image (column 2), the watermarked image (column 3), and the recovered watermark image (column 4). An essential aspect of successful watermarking is the remarkable imperceptibility of the process, as seen by the notable visual resemblance between the original and watermarked images in columns 1 and 3. The recovered watermarks in column 4 show strong fidelity to the original watermarks in column 2, indicating robust extraction even after the embedding process.

FIGURE 10: Green channel watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

Fig. 10 follows an identical layout for the green channel results. This consistent presentation allows for easy comparison with the red channel outcomes. The green channel results exhibit equally impressive imperceptibility, with the watermarked images in column 3 virtually indistinguishable from the originals in column 1. The watermark recovery, shown in column 4, is consistently successful across all test images, further validating our method's effectiveness. The blue channel outcomes are presented in Fig. 11, maintaining the same structured layout as the previous figures. This consistency in presentation facilitates a comprehensive comparison across all three RGB channels. The blue channel results reinforce the effectiveness of our watermarking technique, showcasing consistent performance across the diverse range of test images.

FIGURE 11: Blue channel watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

TABLE 4: Comparison of the RGB color model.

Image	Blue Channel			Green Channel			Red Channel		
	PSNR	SSIM	NC	PSNR	SSIM	NC	PSNR	SSIM	NC
Sailboat	**42.58**	**0.97**	**1.00**	42.46	0.95	1.00	42.16	0.96	1.00
Jetplane	**42.98**	**0.97**	**1.00**	43.03	0.98	0.99	43.29	0.99	1.00
Pepper	**42.55**	**0.98**	**1.00**	42.84	0.97	0.98	42.50	0.95	0.97
Mandrill	**42.23**	**0.98**	**0.99**	42.23	0.98	0.99	42.21	0.97	1.00
Barbara	**36.34**	**0.96**	**0.99**	36.24	0.97	0.99	36.03	0.96	0.97

Table 4 compares red, green, and blue channel quality metrics for five standard test images used in digital watermarking. The analysis employs metrics to evaluate channel suitability, including signal quality, structural similarity, and correlation measures. Comparing across channels, the blue channel consistently demonstrates superior performance. It achieves the highest PSNR and SSIM values for most images, with Sailboat showing a blue channel PSNR of 42.58 dB compared to 42.46 dB and 42.16 dB for green and red channels, respectively. The blue channel also maintains high NC values across all images, indicating a strong watermark correlation. While the green and red channels perform well in specific instances, the blue channel's overall consistency suggests it may be the optimal choice for watermark embedding in many cases. However, after experimenting with the RGB color model, we could not achieve sufficiently high PSNR and SSIM values to meet our imperceptibility goals. This limitation led us to explore alternative color models, ultimately shifting our focus to the YIQ color space. Human visual perception characteristics justify the transition from RGB to YIQ. The YIQ model separates luminance (Y) from chrominance (I and Q), aligning more closely with how the human eye processes visual information. This separation allows for more targeted watermark embedding in components less perceptible to the human eye. The YIQ model offers advantages for watermarking due to its ability to concentrate energy in specific channels, providing a better balance between image quality and watermark robustness.

TABLE 5: Comparison of the YIQ color model.

Image	Y Component			I Component			Q Component		
	PSNR	SSIM	NC	PSNR	SSIM	NC	PSNR	SSIM	NC
Sailboat	58.38	0.99	0.98	**66.57**	**0.99**	**1.00**	73.76	0.99	0.99
Jetplane	55.01	0.99	1.00	**77.02**	**1.00**	**0.99**	80.89	1.00	0.98
Pepper	58.15	0.97	0.92	**66.43**	**0.99**	**1.00**	68.78	0.99	0.99
Mandrill	57.67	0.99	0.97	**66.67**	**0.99**	**1.00**	76.01	1.00	0.98
Barbara	51.85	0.96	0.97	**72.21**	**0.99**	**0.99**	78.46	1.00	0.94

3) Experiment with the YIQ Color Model

The YIQ color model was employed to evaluate the effectiveness of digital watermarking across different image components. Fig. 12,13, and 14, present the results for the "Y", "I", and "Q" components using five standard test images: Sailboat, Jetplane, Pepper, Baboon, and Barbara.

Fig. 12 illustrates the results for the "Y" (luminance) component. The watermark embedding process shows minimal visual impact on the host images, indicating good imperceptibility. The recovered watermarks demonstrate high fidelity, with only slight degradation observed in the Pepper and Barbara images. The "I" (in-phase) component results are presented in Fig. 13. This chrominance component exhibits excellent watermark recovery across all test images. Notably, the Sailboat, Pepper, and Mandrill images show perfect watermark reconstruction, while the others display only marginal degradation.

Fig. 14 showcases the outcomes for the "Q" (quadrature) component. The watermark recovery in this chrominance component is generally good, albeit with slightly more visible degradation than the "I" component. The Barbara image shows the most noticeable watermark degradation among the test sets in this component. Across all three components, the watermarked images remain visually indistinguishable from their original counterparts, suggesting the good imperceptibility of the embedded watermarks. The "I" component offers the most robust watermark recovery, followed closely by the "Y" component, while the "Q" component shows slightly lower but still acceptable performance.

An analysis of the YIQ color space components for five benchmark images shows that the "I" component consistently outperforms the "Y" and "Q" components, as evidenced by the quality metrics presented in Table 5. For example, in the Sailboat image, the "I" component achieves

FIGURE 12: "Y" component watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

FIGURE 13: "I" component watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

a PSNR of 66.57 dB, compared to 58.38 dB for "Y" and 73.76 dB for "Q". The "I" component also shows perfect NC values (NC = 1.00) for Sailboat, Pepper, and Mandrill. In contrast, the "Y" component has NC values of 0.98, 0.92, and 0.97, respectively. Although the "Q" component often has higher PSNR values, its NC scores are generally lower than those of the "I" component. For instance, in the Barbara image, the "Q" component has a PSNR of 78.46 dB and an NC of 0.94, whereas the "I" component has a PSNR of 72.21 dB and an NC of 0.99. In the Jetplane image, the "I" component achieves a perfect SSIM score of 1.00, indicating excellent structural preservation. The "Y" component maintains good SSIM and NC scores across all images but consistently outperforms the "I" component in both imperceptibility (PSNR and SSIM) and robustness (NC).

4) Experiment with Different Size Watermark

The influence of watermark dimensions on the performance of the proposed technique was examined using three distinct sizes: (16x32), (32x64), and (64x128). Fig. 15 illustrates the results for the (16x32) watermark, showing generally good recovery across most test images, with some degradation visible in the Barbara image. Fig. 16 presents the outcomes for the (32x64) watermark, demonstrating improved recovery quality compared to the smaller size, particularly noticeable in the Sailboat and Mandrill images. Fig. 17 showcases the results for the largest (64x128) watermark, exhibiting the most robust recovery across all test images, with near-perfect reconstruction in most cases. Across all three sizes, the watermarked images remain indistinguishable from their original counterparts, indicating good imperceptibility. However, the recovered watermark quality improves as the size increases, suggesting a trade-off between watermark capacity and robustness. This trend is particularly evident when comparing the Barbara image results across Fig. 15,16, and 17, where the larger watermark sizes yield progressively better recovery. The observed trend suggests a positive correlation between watermark size and recovery quality. Watermarks of bigger sizes possess greater redundancy, making them more resilient to distortions caused by the embedding and extraction procedures. This is because more prominent watermarks include a more significant amount of material.

Table 6 presents quality metrics for watermarked images using three different watermark sizes: (16x32), (32x64), and (64x128). Across all test images, a consistent trend emerges as watermark size increases. The (64x128) size consistently yields the highest NC values, indicating superior watermark recovery. For instance, the Sailboat image shows NC im-

(a) (b) (c) (d)

FIGURE 14: "Q" component watermarking analysis. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

(a) (b) (c) (d)

FIGURE 15: Analysis of (16x32) watermark efficacy. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

TABLE 6: Comparative analysis of watermark sizes.

Image	(16x32) Size			(32x64) Size			(64x128) Size		
	PSNR	SSIM	NC	PSNR	SSIM	NC	PSNR	SSIM	NC
Sailboat	74.78	0.99	0.96	72.59	0.99	0.99	**66.57**	**0.99**	**1.00**
Jetplane	86.45	1.00	0.95	82.74	1.00	0.97	**77.02**	**1.00**	**0.99**
Pepper	72.43	0.99	0.97	69.48	0.99	1.00	**66.43**	**0.99**	**1.00**
Mandrill	80.18	1.00	0.93	66.67	0.99	0.99	**66.67**	**0.99**	**1.00**
Barbara	84.23	1.00	0.86	72.21	0.98	0.99	**72.21**	**0.99**	**0.99**

provements from 0.96 (16x32) to 1.00 (64x128). Interestingly, PSNR values decrease as watermark size increases, suggesting a trade-off between watermark robustness and image quality. The Jetplane image demonstrates a notable balance across all metrics, with the (64x128) size achieving an optimal NC of 0.99 while maintaining a high PSNR of 77.02 dB and a perfect SSIM of 1.00. The Barbara image shows the most significant improvement in NC from 0.86 (16x32) to 0.99 (64x128), highlighting the benefits of larger watermarks for complex images. This suggests that images with large uniform areas may benefit, particularly from more prominent watermarks, without significant quality degradation. The Pepper image shows perfect NC (1.00) for both (32x64) and (64x128) sizes, indicating that for some images, a medium-sized watermark may be sufficient to achieve optimal robustness. Notably, SSIM values remain remarkably stable across all watermark sizes, predominantly at 0.99, with instances of 1.00 observed when PSNR exceeds 76 dB. This stability in SSIM suggests that structural similarity is well-preserved regardless of watermark size, even as PSNR decreases. This is particularly important for maintaining the visual quality and integrity of the host image. Overall, the results suggest that the (64x128) watermark size offers the best balance between robustness (as indicated by NC values) and imperceptibility (as shown by high PSNR and SSIM values) across various image types. However, the choice of watermark size may be tailored to specific application requirements, considering the nature of the host image and the desired balance between robustness and imperceptibility.

In exploring activation functions, we investigated the impact on normalized correlation (NC) across five standard images: Sailboat, Jetplane, Pepper, Mandrill, and Barbara. Fig. 18 presents our findings for ReLU, Leaky ReLU, Parametric ReLU, GeLU, SeLU, and ELU. Our results show that ReLU (Fig. 18(a)), Leaky ReLU (Fig. 18(b)), and Parametric ReLU (Fig. 18(c)) consistently yielded high NC values across all datasets, with the majority of scores exceeding 0.95. We observed that Parametric ReLU demonstrated the highest overall performance, with all NC values at or approaching 1.000.

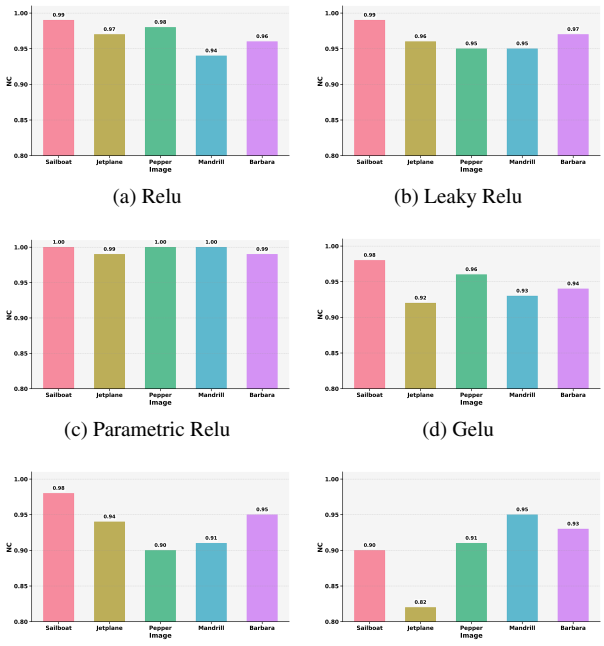
FIGURE 16: Analysis of (32x64) watermark efficacy. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

FIGURE 17: Analysis of (64x128) watermark efficacy. (a) Original images, (b) Watermark images, (c) Watermarked images, (d) Extracted watermark images.

In our experiments with GeLU (Fig. 18(d)) and SeLU (Fig. 18(e)), we found slightly lower but still robust NC, falling within a range of 0.90 to 0.98. The ELU (Fig. 18(f)) exhibited the most variability among the activation functions we tested, with NC values spanning from 0.82 to 0.95. Building upon our activation function analysis, we turned our attention to the effects of principal component analysis (PCA) on normalized correlation (NC), as illustrated in Fig. 19. Our results demonstrate that applying PCA consistently improved NC scores for all tested images. The most significant improvement was observed in the Pepper image, with an NC increase of 0.04. Sailboat, Pepper, and Mandrill images achieved perfect NC scores 1.000 with PCA, while Jetplane and Barbara images reached 0.99. These findings suggest that PCA is an effective preprocessing technique for enhancing image processing outcomes across various image types.

To further investigate PCA's impact, we comprehensively evaluated different PCA dimensionality on normalized correlation (NC) using the same five standard images: Sailboat, Jetplane, Pepper, Mandrill, and Barbara. Our findings for PCA dimensions of 5, 7, 10, and 12 are illustrated in Fig. 20. Our results demonstrate a clear trend of improving NC scores as PCA dimensionality increases. With PCA(5) (Fig. 20(a)), NC values ranged from 0.93 to 0.98. PCA(7) (Fig. 20(b)) showed moderate improvement, with scores between 0.91 and 0.99. A significant jump in performance was observed with PCA(10) (Fig. 20(c)), where all images achieved NC scores of 0.99 or 1.000. PCA(12) (Fig. 20(d)) maintained these high scores, with a slight decrease for the Mandrill and Pepper images. Finally, we explored the relationship between the number of epochs and normalized correlation (NC) for the five standard images: Sailboat, Jetplane, Pepper, Mandrill, and Barbara. The graphs demonstrate a consistent trend of increasing NC values as the number of epochs increases across all images. Sailboat (Fig. 21(a)), Jetplane (Fig. 21(b)), and Pepper (Fig. 21(c)) show similar patterns, with NC values starting around 0.8 at 10 epochs and reaching near or at 1.00 by 50 epochs. Mandrill (Fig. 21(d)) and Barbara (Fig. 21(e)) exhibit slightly lower initial NC values, starting at approximately 0.75-0.78 at 10 epochs, but still achieve high NC scores of 1.00 or nearly 1.00 by 50 epochs. All images display rapid improvement in NC between 10 and 30 epochs, followed by a more gradual increase or plateau from 30 to 50. This suggests that most performance gains occur within the first 30 training epochs.

We evaluated a range of hyperparameters for training our extraction model, as detailed in Table 7. Based on our studies, we used the ADAM optimizer with a learning rate of 0.001



(a) Relu (b) Leaky Relu

(c) Parametric Relu (d) Gelu

(e) Selu (f) Elu

FIGURE 18: Comparative analysis of activation functions on NC.

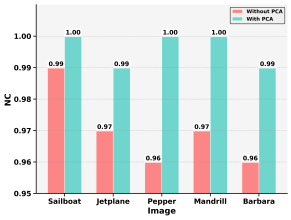


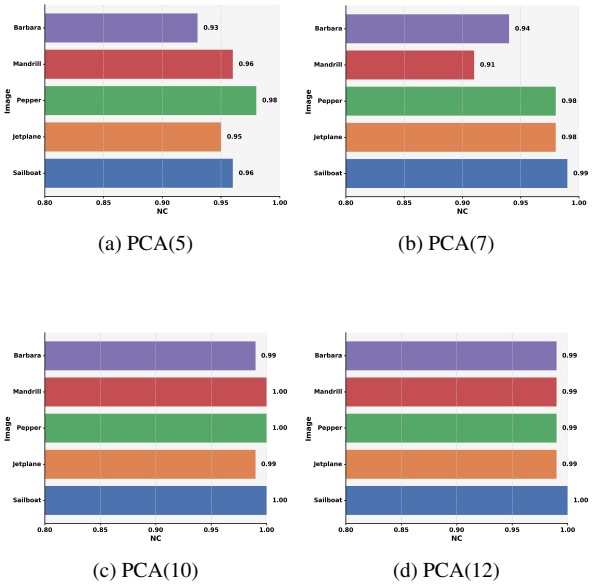
FIGURE 19: Comparison of NC with and without PCA.

TABLE 7: Hyperparameters for the extraction model.

Hyperparameters	Extraction Framework
Learning rate	0.001
Loss	Mean Squared Error
Epochs	50
Optimizer	ADAM
Beta 1	0.90
Beta 2	0.99

and beta values of 0.90 and 0.99. We selected Mean Squared Error as our loss function and determined that training the model for 50 epochs yielded optimal results. Through rigorous testing, we found that this specific combination of hyperparameters ensures robust performance and efficient convergence of the extraction model during the training phase.

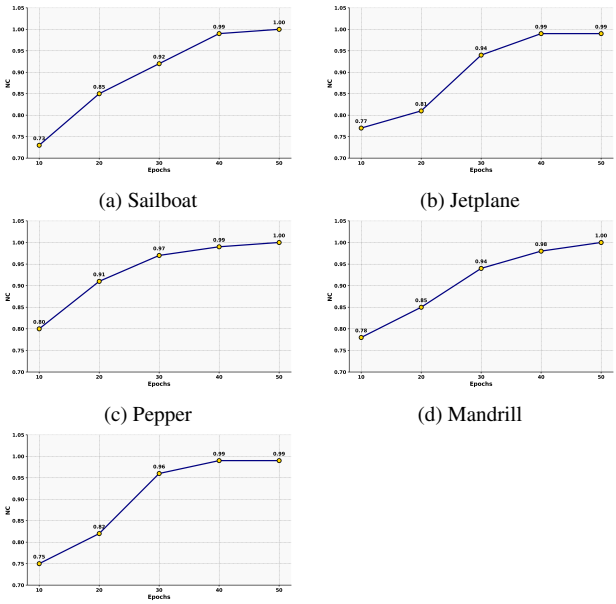
We conducted a series of comprehensive experiments to assess the durability of our watermarking process. First, we evaluated the normalized correlation (NC) values of extracted watermarks after subjecting them to various image processing attacks on different test images in Table 8. This analysis demonstrated our method's resilience against common manipulations. We then examined the bit error rate (BER) of extracted watermarks under similar attack conditions in Table 9, providing insights into the accuracy of watermark extraction under different distortions. Finally, we tested our technique's performance against more complex scenarios by applying combined attacks and measuring the resulting NC values in Table 10. This last experiment showcased our method's ability to withstand sophisticated image manipulations.



(a) PCA(5) (b) PCA(7)

(c) PCA(10) (d) PCA(12)

FIGURE 20: Impact of PCA dimensionality on NC.



(a) Sailboat (b) Jetplane

(c) Pepper (d) Mandrill

(e) Barbara

FIGURE 21: Epochs vs NC for five standard images. The graphs demonstrate how the NC value improves as the number of epochs increases.

TABLE 8: NC of the extracted watermarks under various attacks.

Attack	Sailboat	Jetplane	Pepper	Mandrill	Barbara
No Attack	1	0.9981	1	1	0.9992
Speckle Noise (0.01)	1	0.9572	0.9973	0.9238	0.9822
Sharpening (0.01)	1	0.9852	1	0.9527	0.9976
Rotation (10)	0.7881	0.6253	0.7085	0.5829	0.6986
Salt & Pepper Noise (0.01)	0.9914	0.9553	0.9984	0.9781	0.9423
Average Filtering (3x3)	0.5301	0.6157	0.5702	0.5672	0.4560
Wiener Filtering (3x3)	0.7385	0.6605	0.7082	0.6055	0.5260
Median Filtering (3x3)	0.9057	0.8852	0.7702	0.6618	0.5442
Gaussian Noise	0.9170	0.9542	0.9097	0.9635	0.9352
Histogram Equalization (0.5)	1	0.9992	1	0.9842	0.9752
Brightness Adjustment (1.5)	1	0.9775	1	0.9380	0.9575
Gamma Correction (0.5)	0.9798	0.9588	0.9905	0.9255	0.9997
JPEG (QF=90)	0.9580	0.9779	0.9998	0.9097	0.8875
Cropping (10%)	0.5377	0.4850	0.4397	0.5708	0.6177
Gaussian Blur	0.7769	0.6253	0.7056	0.6057	0.6241

TABLE 9: BER in watermark extraction after image manipulations.

Attacks	Sailboat	Jetplane	Pepper	Mandrill	Barbara
No Attack	0.0318	0.0399	0.0969	0.0510	0.0439
Speckle Noise (0.01)	0.0592	0.0902	0.0642	0.0691	0.1047
Sharpening (0.01)	0.0943	0.1119	0.0846	0.0950	0.1482
Rotation (10)	0.1872	0.1873	0.1895	0.1927	0.2067
Salt & Pepper Noise (0.01)	0.0675	0.0665	0.0619	0.0720	0.0898
Average Filtering (3x3)	0.0319	0.0398	0.0971	0.0509	0.0439
Weiner Filtering (3x3)	0.0319	0.0398	0.0971	0.0509	0.0439
Median Filtering (3x3)	0.1937	0.1994	0.2058	0.2062	0.2379
Gaussian Noise	0.0679	0.0847	0.0622	0.0682	0.1097
Histogram Equalization (0.5)	0.0762	0.0988	0.1177	0.0528	0.1218
Brightness Adjustment (1.5)	0.9930	0.9750	0.9940	0.9370	0.9560
Gamma Correction (0.5)	0.1302	0.1353	0.1777	0.0708	0.1640
JPEG (QF=90)	0.1540	0.1446	0.1521	0.1434	0.1625
Cropping (10%)	0.1166	0.1271	0.1295	0.0764	0.1082
Gaussian Blur	0.7750	0.6240	0.7045	0.6044	0.6230

TABLE 10: NC values for various combined attack scenarios.

Attack Scenarios	NC Value
10% Crop + JPEG (QF=90) Compression	0.5972
JPEG (QF=90) Compression + Salt & Pepper Noise (0.02)	0.8905
Salt & Pepper Noise (0.02) + 10% Crop	0.8165
Poison Noise + JPEG (QF=90) Compression	0.9280
Histogram Equalization + Salt & Pepper Noise (0.02)	0.9920
JPEG (QF=90) Compression + 10% Crop	0.6975
Horizontal Flip + Salt & Pepper Noise (0.02)	0.9938
10Rotation + Histogram Equalization	0.5750
Sharpening + JPEG (QF=90) Compression	0.9570
Speckle Noise (0.05) + Histogram Equalization	0.9835
Vertical Flip + 10% Crop	0.5882
Histogram Equalization + JPEG (QF=90) Compression	0.9992
Gamma Correction (0.5) + Gaussian Noise (0.05)	0.8840

E. RATIONALE FOR SELECTING BENCHMARK METHODS

To ensure a comprehensive evaluation, we selected MUSIC, SVR, and DNN as benchmark methods for comparison in this study. These methods were chosen based on their relevance and performance in related domains, as detailed below:

- **MUSIC (Multiple Signal Classification):** MUSIC is a widely used spectral estimation algorithm renowned for its high resolution in signal processing applications. It serves as a strong baseline due to its robustness in noise scenarios and ability to resolve closely spaced components effectively.
- **SVR (Support Vector Regression):** SVR is a supervised learning method well-suited for regression problems, particularly in nonlinear data scenarios. It was selected for its ability to provide accurate results with minimal overfitting, which is critical for tasks involving small or medium-sized datasets.
- **DNN (Deep Neural Network):** DNN represents state-of-the-art machine learning, known for its capacity to model complex and high-dimensional relationships. It

was included as a benchmark to leverage its superior feature representation and predictive power compared to traditional algorithms.
- **Proposed Method:** The proposed method integrates the Catalan transform, non-negative matrix factorization (NMF), and artificial neural networks (ANN). This synergy achieves high imperceptibility, robustness, and computational efficiency, making it a novel contribution to watermarking techniques.

Table 11 provides a brief comparison of the selected methods, highlighting their key features, advantages, and limitations.

TABLE 11: Summary of rationale for benchmark methods.

Method	Rationale
MUSIC	Classical method for high-resolution spectral estimation. Provides a traditional non-machine-learning baseline.
SVR	Machine-learning approach for handling non-linear relationships. Effective in watermarking tasks.
DNN	Modern deep-learning approach for feature extraction and recognition. Highlights the advantages of advanced learning techniques.
Proposed Method	Combines Catalan transform and NMF with neural network-based learning. Ensures high PSNR and SSIM.

Including these methods ensures a balanced evaluation by considering classical techniques (MUSIC, SVR) and modern deep learning-based approaches (DNN). Each method was selected to represent a distinct approach to solving the problem, providing insights into their relative performance and trade-offs.

TABLE 12: Evaluation of the proposed scheme in comparison to state-of-the-art methods.

	PSNR (dB)	SSIM
[39]	22.46 to 36.16	0.98 to 0.99
[29]	27.57 to 45.28	—
[61]	31.05 to 39.70	0.93 to 0.97
[47]	32.28 to 38.61	—
[60]	33.68 to 38.49	0.93 to 0.96
[41]	34.72 to 37.89	0.97 to 0.98
[40]	35.22 to 37.86	—
[44]	35.41 to 39.88	0.96 to 0.97
[45]	36.93 to 40.76	0.96 to 0.99
[43]	39.73 to 40.25	0.97 to 0.99
[25]	50.39 to 53.77	—
[46]	50.48 to 59.22	—
[48]	53.30 to 62.88	0.98 to 0.99
Proposed	**66.43 to 77.02**	**0.99 to 1.00**

We comprehensively analyzed our proposed image protection technique, comparing its performance against various state-of-the-art methods. Our evaluation focused on two crucial aspects: perceptual quality and robustness against attacks. Table 12 showcases the perceptual quality metrics (PSNR and SSIM) for different image protection techniques. Our proposed method exhibited exceptional performance, achieving the highest PSNR range of (66.43 to 77.02) dB. This result significantly surpasses other state-of-the-art approaches, with the next best result being (53.30 to 62.88) dB [48]. Most methods, including ours, attained high SSIM values, with our method achieving the highest range of (0.99 to 1.00), indicating excellent structural similarity to the original image. We observed that some earlier techniques [39], [40], [41], showed lower PSNR values in the (22 to 38) dB range, while more recent approaches [25], [46], [48], demonstrated improved performance with PSNR values above 50 dB. Among these, [60] and [61] introduced innovative strategies using sparse attention and transformer-based models, achieving PSNR ranges of (33.68 to 38.49) dB and (31.05 to 39.70) dB, respectively. Despite their advancements, their results still fall short compared to our method, which not only achieves the highest PSNR values but also maintains exceptional SSIM metrics. To complement our perceptual quality analysis, we extensively evaluated robustness against various attacks, as presented in Table 13. Our proposed method demonstrated outstanding resilience across a wide range of distortions. For noise attacks (Salt & pepper, Gaussian, Speckle), we maintained high normalized correlation (NC) values, often outperforming other techniques. Our method showed particular strength against salt & pepper noise, with ($NC > 0.98$) even at high noise levels. We achieved perfect watermark extraction ($NC = 1$) under geometric and processing attacks like sharpening and histogram equalization, surpassing all compared methods. In the case of median filtering, our approach ($NC = 0.9057$) outperformed several recent techniques like [45] ($NC = 0.6997$) and [47] ($NC = 0.6827$). While our method showed some vulnerability to cropping ($NC = 0.5377$ for 10% crop), we noted that this is a common challenge for many watermarking schemes. Under JPEG compression (QF = 90%), our method (NC = 0.9580) performed comparably to other robust methods. The results from both tables demonstrate a strong correlation between our method's high perceptual quality and its robustness. The exceptional PSNR achieved by our technique, as shown in Table 12, translates to superior resilience against various attacks, as evidenced in Table 13.

1) Security Analysis of Watermark

The relative entropy (RE) quantifies the dissimilarity in variety between two probability distributions when comparing two digital image histograms. The relative entropy (RE) of two probability distributions can be calculated by adding all the potential outcomes. This is denoted as:

$$R(I) = \sum_{i=1}^{I} w(I_i) \log_2 \frac{w(I_i)}{w'(I_i)} \tag{18}$$

TABLE 13: Comparative analysis of the suggested scheme's robustness in comparison to state-of-the-art.

	[39]	[40]	[41]	[29]	[43]	[44]	[45]	[47]	[25]	[46]	[48]	**Proposed**
No Attack	1	0.9189	0.9822	0.9976	1	0.9967	1	0.9987	0.9960	0.9979	0.9934	1
Speckle Noise (0.01)	0.9968	—	0.9820	0.8697	0.998	0.9276	0.9993	—	0.9648	—	0.8301	1
Speckle Noise (0.02)	0.9895	—	—	—	0.984	—	0.9989	—	0.9140	—	—	0.9987
Speckle Noise (0.05)	0.9046	—	—	—	0.863	—	0.9965	—	—	—	—	0.9866
Gaussian Noise (0.01)	0.9966	—	0.9788	0.8201	0.962	0.7996	0.9970	0.8489	—	—	0.8243	0.9578
Gaussian Noise (0.02)	0.9868	—	—	0.770	—	—	—	—	—	0.8225	—	0.9170
Gaussian Noise (0.05)	0.9072	—	—	0.8379	0.438	—	0.9824	—	0.7461	—	—	0.8824
Salt & Pepper Noise (0.01)	0.9988	0.9847	0.9813	0.9025	0.994	0.9253	0.9993	0.9829	0.9476	0.9477	0.8764	0.9994
Salt & Pepper Noise (0.02)	—	—	0.8855	—	0.946	0.8693	0.9981	—	0.8007	0.9170	—	0.9951
Salt & Pepper Noise (0.05)	0.9967	—	0.9885	—	0.789	—	0.9975	—	—	—	—	0.9877
Sharpening	—	—	—	0.8560	—	—	0.9920	—	—	—	0.9250	1
Median Filtering (3×3)	0.9420	0.6789	—	0.8271	0.978	0.9269	0.6997	0.6827	—	0.8175	—	0.9057
Histogram Equalization	—	—	0.9821	—	0.8812	—	0.9997	—	—	—	—	1
Poison Noise	—	—	—	—	0.99	—	0.9996	0.9770	—	—	—	0.9076
JPEG Compression (QF=90%)	0.9921	0.9901	—	0.9479	0.99	0.9948	0.9989	—	—	0.9655	0.8629	0.9580
Crop 10%	—	—	—	—	—	—	0.9997	—	0.0937	—	—	0.5377

Here, the probability distributions are denoted by $w(I_i)$ and $w'(I_i)$. If the two distributions $w(I_i)$ are identical, then the relative entropy will equal 0. The proposed methodology is evaluated against other conventional methods [49], [50] in Table 14. Based on the statistics presented in the table, the suggested scrambling strategy demonstrates superior security compared to the other options, as it yields higher RE values.

TABLE 14: Comparison of the proposed algorithm with related works based on RE.

Methods of scrambling	RE values
[49]	0.2211
[50]	0.2135
Proposed method	**0.3094**

The unified average change intensity (UACI) and the number of pixel change rate (NPCR) metrics can be used to assess the robustness of a scrambling algorithm, as mentioned in [55]. The NPCR calculates the percentage of affected pixels in the corrupted image for every changed original pixel. The original and scrambled image pixel values are represented by the symbols $x(i,j)$ and $x'(i,j)$ in the i-th row and j-th column, respectively. Here is how NPCR is defined:

$$\text{NPCR} = \frac{\sum_{i,j} W(i,j)}{I \times J} \times 100\% \tag{19}$$

where:

$$W(i,j) = \begin{cases} 0 & \text{if } x(i,j) = x'(i,j) \\ 1 & \text{if } x(i,j) \neq x'(i,j) \end{cases} \tag{20}$$

According to the research conducted in [51], a greater NPCR value indicates a more significant impact on the image's appearance due to pixel alterations. The theoretical, critical values are above 90%. By comparing the original and scrambled images, UACI measures the average intensity difference and is computed as:

$$\text{UACI} = \frac{1}{I \times J} \sum_{i,j} \frac{|w(i,j) - w'(i,j)|}{255} \times 100\% \tag{21}$$

Above 33%, there are theoretical, critical values for UACI [51]. A UACI of 34.21% and an NPCR of 99.15% are achieved by the suggested technique. Table 15, compares the NPCR and UACI values of the proposed method to those of two conventional methods. The outcomes show that the proposed method achieves NPCR values of 98.48% and 98.76%, and UACI values of 25.40 and 24.38, respectively.

TABLE 15: Assessment of the proposed technique compared to prior studies using NPCR and UACI.

Scrambling methods	NPCR (%)	UACI (%)
[49]	98.48	25.40
[50]	98.76	24.38
Proposed method	**99.15**	**34.21**

2) Payload Analysis

An essential factor in watermarking is the payload, which quantifies the amount of data that can be embedded into the cover image [52]. The scholarly investigation employs a host image with dimensions of $(512 \times 512 \times 3)$ and a watermark with dimensions of (64×128). The payload for the proposed method is calculated as follows:

$$\begin{aligned} \text{Data payload} &= \frac{\text{Number of watermark bits}}{\text{Number of host image pixels}} \\ &= \frac{64 \times 128}{512 \times 512} \\ &= 0.03125 \text{ bpp} \end{aligned} \tag{22}$$

TABLE 16: Comparison of data payload between the proposed method and state-of-the-art methods.

Literature	Data payload (bpp)
[45]	0.04166
[53]	0.0625
Proposed	**0.03125**

Table 16 shows a comparison of the data payload between the proposed method and state-of-the-art methods [45] and [53]. The proposed and the state-of-the-art methods achieve a data payload of 0.03125 bpp, 0.04166 bpp, and 0.0625 bpp, respectively. In other words, the proposed method provides a slightly lower data payload than the method reported in [45] and offers a comparable data payload to the method reported in [53], demonstrating a reasonably competitive embedding capacity.

3) Analysis of Time Complexity

The temporal complexity of applying the Catalan transform (CT) and non-negative matrix factorization (NMF) on an image of size ($n \times n$), which is divided into sub-blocks of size (2×2), is examined. The process can be broken down as follows:

1) **Image subdivision:** The number of sub-blocks is given by:

$$\text{Number of sub-blocks} = \frac{n \times n}{2 \times 2} = \frac{n^2}{4} \tag{22}$$

2) **Catalan transform:** Applied to each 2×2 sub-block:
   - Time complexity for one sub-block: $\mathcal{O}(1)$
   - Total time complexity: $T_{Catalan} = \frac{n^2}{4} \times \mathcal{O}(1) = \mathcal{O}(n^2)$
3) **Non-negative matrix factorization (NMF):** Applied to each transformed sub-block:
   - Time complexity for one sub-block: $\mathcal{O}(1)$
   - Total time complexity: $T_{NMF} = \frac{n^2}{4} \times \mathcal{O}(1) = \mathcal{O}(n^2)$
4) **Overall time complexity:** Sum of the constituent parts:

$$T_{total} = T_{Catalan} + T_{NMF} = \mathcal{O}(n^2) + \mathcal{O}(n^2) = \mathcal{O}(n^2) \tag{23}$$

TABLE 17: Comparative analysis of time complexity across various watermarking algorithms.

Literature	Time Complexity
DWT-WHT-SVD [45]	$\mathcal{O}(n^3)$
DWT-SVD [48]	$\mathcal{O}(n^3)$
FrHT+DT-CWT+SVD [54]	$\mathcal{O}(n^3)$
DWT-DCT [20]	$\mathcal{O}(n^2 \log(n))$
Catalan-NMF (**Proposed**)	$\mathcal{O}(n^2)$

The analysis reveals that the Catalan-NMF transform has a quadratic time complexity of $\mathcal{O}(n^2)$ concerning the image dimension. Table 17 illustrates the time complexity of various watermarking algorithms. The proposed Catalan-NMF method achieves $\mathcal{O}(n^2)$, showing improved efficiency compared to existing techniques. Most prior methods, such as DWT-WHT-SVD and FrHT+DT-CWT+SVD, exhibit $\mathcal{O}(n^3)$ complexity, while DWT-DCT performs at $\mathcal{O}(n^2 \log(n))$. This comparison demonstrates the computational advantage of our proposed approach in image watermarking applications.

4) Trade-off Analysis

Efficient watermarking requires a careful balance between imperceptibility, security, and resilience. To be effective, procedures must meet specific requirements: With a PSNR of more than 35 dB, the watermarked image is very similar to the original and has excellent perceptual accuracy. The number of pixel change rate (NPCR) and the unified average changing intensity (UACI) are the leading indicators used to ensure security and guard against statistical assaults. A UACI more than 33% shows considerable differences in pixel values between the original and watermarked images, while an NPCR over 90% indicates massive pixel changes. The resilience of a watermark against various image processing techniques and attacks is determined by the normalized correlation (NC) values, which are commonly used as a metric. A number near 1 indicates a high resistance to these processes. Optimizing all parameters simultaneously can be challenging; for instance, enhancing the imperceptibility of the watermark may compromise its strength, while improving its security may result in a decrease in visual quality.

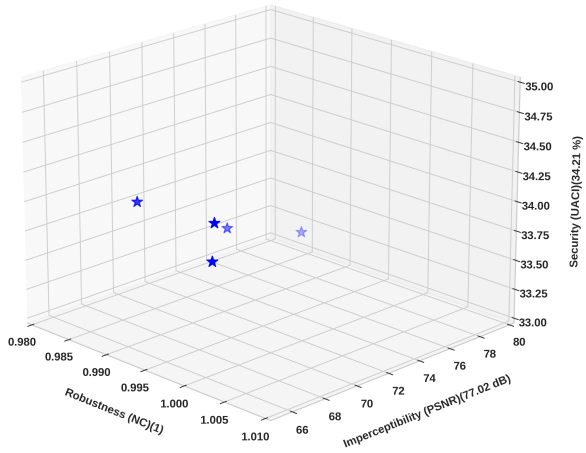


FIGURE 22: Trade-off among robustness, security (UACI), and imperceptibility.

A UACI more than 33% shows considerable differences in pixel values between the original and watermarked images, while an NPCR over 90% indicates huge changes in pixels. The resilience of a watermark against various image processing techniques and attacks is determined by the normalized correlation (NC) values, which are commonly used as a metric. A number near 1 indicates a high resistance to these processes. Optimizing all parameters simultaneously can be challenging; for instance, enhancing the imperceptibility of

the watermark may compromise its strength, while enhancing its security may result in a decrease in visual quality. Fig. 22 illustrates the trade-off among these three parameters, with the X-axis representing robustness, the Y-axis representing imperceptibility (measured by PSNR), and the Z-axis representing security (measured by UACI). Our proposed method shows an imperceptibility of over 66 dB, average robustness (NC value) exceeding 99%, and a UACI value of 34.21%.

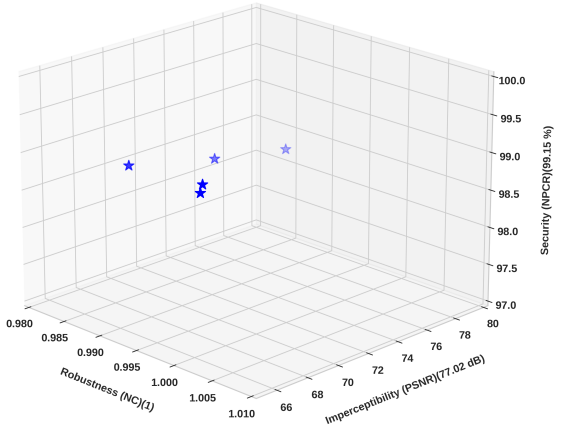


FIGURE 23: Trade-off among robustness, security (NPCR), and imperceptibility.

Fig. 23 further explores this trade-off, maintaining the same imperceptibility and robustness values while varying the security parameter (measured by NPCR). Here, the X-axis represents robustness, the Y-axis represents imperceptibility (PSNR), and the Z-axis represents security (NPCR). The results demonstrate an NPCR value of 99.15%, alongside the same PSNR and NC values, indicating that our method achieves high levels of imperceptibility, security, and robustness, thereby maintaining an optimal trade-off among these factors.

F. DISCUSSION

The proposed watermarking methodology ensures very high imperceptibility, robustness, and efficiency. Given the possible questions that may arise, adding an ANN in the extraction phase enhances robustness and accuracy by a significant degree due to its powerful ability to model complex nonlinear relations within statistical features. ANNs handle all kinds of distortions, such as compression and noise, well, ensuring adaptability across diverse scenarios. Although other variants, such as CNNs or transformers, perform well on spatial or sequential data, ANNs better suit structured, tabular data used during the extraction phase. Again, deep learning is explicit in the embedding phase to maintain a careful balance between traditional embedding techniques, including the Catalan transform and non-negative matrix factorization, with the adaptability granted by deep learning. This thoughtfulness in design forms a new synergy, drawing strengths from traditional and contemporary approaches while preserving the unique contributions of the Catalan transform and NMF. Despite these advantages, the methodology also has some drawbacks that call for attention:

1. **Reduced robustness to geometric attacks:** While the method performs well under standard attacks such as noise addition and compression, its robustness against geometric distortions like rotation and scaling is limited. Incorporating geometric invariant transforms or enhancing the ANN with augmented training data could improve performance in this area.
2. **Limited data payload:** The methodology prioritizes robustness and imperceptibility, leading to a lower data payload due to the use of a binary watermark. Future work could explore increasing the data payload by utilizing grayscale or color watermarks, as they allow embedding more information per pixel without significantly compromising quality.
3. **Dependency on parameter tuning:** The proposed method is sensitive to accurate parameter tuning, including threshold values and ANN training parameters. The use of automated optimization techniques, such as genetic algorithms or reinforcement learning, would resolve this problem.
4. **Moderate performance under extreme noise:** While resistant to medium-level noise, this method still faces challenges with very heavy noise. Future improvements may include the integration of advanced denoising techniques or hybrid models to increase robustness.

By overcoming these limitations, the proposed method could be more applicable and achieve better performance in various scenarios, further solidifying its utility in digital watermarking.

VI. CONCLUSION

Synthesizing our findings, this study introduces a new watermarking methodology for protecting digital images. In particular, our approach to achieving protection against unauthorized duplication and distribution of digital content in favor of copyright protection exploits encryption techniques and watermarking algorithms. The main key findings of the proposed method are higher PSNR and SSIM values, which indicate a more significant value of a greater value increase of NC, less time complexity, higher payload, and enhanced watermark security. With thorough testing and evaluation, these enhancements make our approach unobtrusive, robust, and resistant to many attack types. This methodology responds to the urgent need in today's landscape for an increased level of protection for digital assets, affording a strong and effective tool to content creators, distributors, and rights holders in the protection of intellectual property rights and maintenance of the integrity of their works. It is expected that this research will further open the doors to more innovations in this field and bring in a secured digital ecosystem. Future work should focus on extending the methodology's applicability to grayscale and color images, improving resilience against rotation and median filtering attacks, and integrating advanced encryption techniques. Additionally, exploring

alternative deep learning architectures and developing efficient algorithms for real-time applications and low-latency environments will further enhance watermarking technology, building upon the achievements in processing speed and data payload demonstrated in this study.

REFERENCES

[1] A. Z. Tirkel, G. A. Rankin, R. M. Van Schyndel, W. J. Ho, N. R. A. Mee, and C. F. Osborne, "Electronic watermark," in *Digital Image Computing, Technology and Applications (DICTA'93)*, 1993, pp. 666–673.

[2] H. T. Sencar, M. Ramkumar, and A. N. Akansu, *Data Hiding Fundamentals and Applications: Content Security in Digital Multimedia*, Amsterdam: Elsevier, 2004.

[3] R. B. Wolfgang, C. I. Podilchuk, and E. J. Delp, "Perceptual watermarks for digital images and video," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1108–1126, 1999.

[4] O. P. Singh, A. K. Singh, G. Srivastava, and N. Kumar, "Image watermarking using soft computing techniques: A comprehensive survey," *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30367–30398, 2021.

[5] P. Amrit and A. K. Singh, "Survey on watermarking methods in the artificial intelligence domain and beyond," *Computer Communications*, vol. 188, pp. 52–65, 2022.

[6] C. Kumar, A. K. Singh, and P. Kumar, "A recent survey on image watermarking techniques and its application in e-governance," *Multimedia Tools and Applications*, vol. 77, pp. 3597–3622, 2018.

[7] H. K. Singh and A. K. Singh, "Comprehensive review of watermarking techniques in deep-learning environments," *Journal of Electronic Imaging*, vol. 32, no. 3, pp. 031804–031804, 2023.

[8] S. P. Mohanty, A. Sengupta, P. Guturu, and E. Kougianos, "Everything you want to know about watermarking: From paper marks to hardware protection," *IEEE Consumer Electronics Magazine*, vol. 6, no. 3, pp. 83–91, 2017.

[9] A. Anand and A. K. Singh, "A comprehensive study of deep learning-based covert communication," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 18, no. 2s, pp. 1–19, 2022.

[10] P. Bas, B. Roue, and J.-M. Chassery, "Tatouage d'images couleur additif: vers la sélection d'un espace d'insertion optimal," *Coresa03, Lyon, France*, 2003.

[11] C.-H. Chou and K.-C. Liu, "A perceptually tuned watermarking scheme for color images," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2966–2982, 2010.

[12] C. Das, S. Panigrahi, V. K. Sharma, and K. K. Mahapatra, "A novel blind robust image watermarking in DCT domain using inter-block coefficient correlation," *AEU-International Journal of Electronics and Communications*, vol. 68, no. 3, pp. 244–253, 2014.

[13] J. Liu, J. Huang, Y. Luo, L. Cao, S. Yang, D. Wei, and R. Zhou, "An optimized image watermarking method based on HD and SVD in DWT domain," *IEEE Access*, vol. 7, pp. 80849–80860, 2019.

[14] A. Al-Haj, "Combined DWT-DCT digital image watermarking," *Journal of Computer Science*, vol. 3, no. 9, pp. 740–746, 2007.

[15] M. Cheng, L. Yan, Y. Zhou, and M. Lei, "A combined DWT and DCT watermarking scheme optimized using genetic algorithm," *Journal of Multimedia*, vol. 8, no. 3, pp. 299–305, 2013.

[16] M. A. Jemal, M. Mziou Sallami, and F. Ghorbel, "Robust watermarking method based on the analytical Clifford Fourier Mellin transform," *Multimedia Tools and Applications*, vol. 83, no. 9, pp. 25901–25922, 2024.

[17] B. Singh and M. K. Sharma, "Watermarking technique for document images using discrete curvelet transform and discrete cosine transform," *Multimedia Tools and Applications*, pp. 1–25, 2024.

[18] S. A. Hosseini and P. Farahmand, "An attack resistant hybrid blind image watermarking scheme based on combination of DWT, DCT and PCA," *Multimedia Tools and Applications*, vol. 83, no. 7, pp. 18829–18852, 2024.

[19] S. Kumar, S. Verma, B. K. Singh, V. Kumar, S. Chandra, and C. Barde, "Entropy based adaptive color image watermarking technique in YCbCr color space," *Multimedia Tools and Applications*, vol. 83, no. 5, pp. 13725–13751, 2024.

[20] S. Sharma, H. Sharma, and J. B. Sharma, "Artificial bee colony based perceptually tuned blind color image watermarking in hybrid LWT-DCT domain," *Multimedia Tools and Applications*, vol. 80, no. 12, pp. 18753–18785, 2021.

[21] J. Chen, J. Zhang, K. Debattista, and J. Han, "Semi-supervised unpaired medical image segmentation through task-affinity consistency," *IEEE Transactions on Medical Imaging*, vol. 42, no. 3, pp. 594–605, 2022.

[22] Y. Liu, D. Zhang, Q. Zhang, and J. Han, "Part-object relational visual saliency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3688–3704, 2021.

[23] R. Radha Kumari, V. Vijaya Kumar, and K. Rama Naidu, "Deep learning-based image watermarking technique with hybrid DWT-SVD," *The Imaging Science Journal*, pp. 1–17, 2023.

[24] H. K. Singh and A. K. Singh, "Digital image watermarking using deep learning," *Multimedia Tools and Applications*, vol. 83, no. 1, pp. 2979–2994, 2024.

[25] S. Jaiswal and M. K. Pandey, "Deep Artificial Neural Network Based Blind Color Image Watermarking," in *Doctoral Symposium on Human Centered Computing*, 2023, pp. 101–112.

[26] X. Guo, W. Yang, L. Zhang, Y. Shi, J. Li, J. Sun, and W. Wan, "Deep image watermarking with loss-driven modification," *Multimedia Tools and Applications*, vol. 83, no. 13, pp. 37665–37685, 2024.

[27] Q. Wei, H. Wang, and G. Zhang, "A robust image watermarking approach using cycle variational autoencoder," *Security and Communication Networks*, vol. 2020, no. 1, pp. 8869096, 2020.

[28] S. Mukhopadhyay, S. Hossain, S. K. Ghosal, and R. Sarkar, "Secured image steganography based on Catalan transform," *Multimedia Tools and Applications*, vol. 80, pp. 14495–14520, 2021.

[29] Z. Hua, S. Yi, and Y. Zhou, "Medical image encryption using high-speed scrambling and pixel adaptive diffusion," *Signal Processing*, vol. 144, pp. 134–144, 2018.

[30] M. Dua, A. Kumar, A. Garg, and V. Garg, "Multiple image encryption approach using non linear chaotic map and cosine transformation," *International Journal of Information Technology*, vol. 14, no. 3, pp. 1627–1641, 2022.

[31] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ: Pearson, 2009.

[32] B. Jähne, *Digital Image Processing*, 6th ed. Berlin, Heidelberg: Springer, 2005.

[33] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York, NY: McGraw-Hill, 2002.

[34] D. Zheng, B. Zhang, and R. Chalaturnyk, "Uncertainty quantification of in situ horizontal stress with pressuremeter using a statistical inverse analysis method," *Canadian Geotechnical Journal*, vol. 59, no. 3, pp. 397–409, 2022.

[35] D. D. Wackerly, W. Mendenhall, and R. L. Scheaffer, *Mathematical Statistics with Applications*, 7th ed. Belmont, CA: Brooks/Cole, 2008.

[36] L. Ma and S. Song, "Improved image watermarking scheme using non-negative matrix factorization and wavelet transform," in *2009 International Conference on Wireless Communications & Signal Processing*, 2009, pp. 1–5.

[37] "USC-SIPI Image Database," [Online]. Available: https://sipi.usc.edu/database/database.php?volume=textures&image=34. [Accessed: 5 July 2023].

[38] "CCIA Image Database," [Online]. Available: https://ccia.ugr.es/cvg/dbimagenes/. [Accessed: 5 July 2023].

[39] C. Mouli PVSSR, "A robust semi-blind watermarking for color images based on multiple decompositions," *Multimedia Tools and Applications*, vol. 76, pp. 25623–25656, 2017.

[40] F. Zhang, T. Luo, G. Jiang, M. Yu, H. Xu, and W. Zhou, "A novel robust color image watermarking method using RGB correlations," *Multimedia Tools and Applications*, vol. 78, pp. 20133–20155, 2019.

[41] F. Yasmeen and M. S. Uddin, "An efficient watermarking approach based on LL and HH edges of DWT–SVD," *SN Computer Science*, vol. 2, no. 2, pp. 82, 2021.

[42] S. Roy and A. K. Pal, "An SVD based location specific robust color image watermarking scheme using RDWT and Arnold scrambling," *Wireless Personal Communications*, vol. 98, pp. 2223–2250, 2018.

[43] R. Sinhal, D. K. Jain, and I. A. Ansari, "Machine learning based blind color image watermarking scheme for copyright protection," *Pattern Recognition Letters*, vol. 145, pp. 171–177, 2021.

[44] H. Zhang, Z. Li, X. Liu, C. Wang, and X. Wang, "Robust image watermarking algorithm based on QWT and QSVD using 2D Chebyshev-Logistic map," *Journal of the Franklin Institute*, vol. 359, no. 2, pp. 1755–1781, 2022.

[45] S. Kumar, S. Verma, B. K. Singh, V. Kumar, S. Chandra, and C. Barde, "Entropy based adaptive color image watermarking technique in YCbCr color space," *Multimedia Tools and Applications*, vol. 83, no. 5, pp. 13725–13751, 2024.

[46] S. A. Hosseini and P. Farahmand, "An attack resistant hybrid blind image watermarking scheme based on combination of DWT, DCT and PCA," *Multimedia Tools and Applications*, vol. 83, no. 7, pp. 18829–18852, 2024.
[47] X. Guo, W. Yang, L. Zhang, Y. Shi, J. Li, J. Sun, and W. Wan, "Deep image watermarking with loss-driven modification," *Multimedia Tools and Applications*, vol. 83, no. 13, pp. 37665–37685, 2024.
[48] R. Radha Kumari, V. Vijaya Kumar, and K. Rama Naidu, "Deep learning-based image watermarking technique with hybrid DWT-SVD," *The Imaging Science Journal*, pp. 1–17, 2023.
[49] Q. Su, Z. Yuan, and D. Liu, "An approximate Schur decomposition-based spatial domain color image watermarking method," *IEEE Access*, vol. 7, pp. 4358–4370, 2018.
[50] Q. Su, G. Wang, X. Zhang, G. Lv, and B. Chen, "A new algorithm of blind color image watermarking based on LU decomposition," *Multidimensional Systems and Signal Processing*, vol. 29, pp. 1055–1074, 2018.
[51] Q. Su, G. Wang, X. Zhang, G. Lv, and B. Chen, "A new algorithm of blind color image watermarking based on LU decomposition," *Multidimensional Systems and Signal Processing*, vol. 29, pp. 1055–1074, 2018.
[52] S. Kumar and B. K. Singh, "An improved watermarking scheme for color image using alpha blending," *Multimedia Tools and Applications*, vol. 80, pp. 13975–13999, 2021.
[53] H. Wang, X. Tian, and Y. Xia, "A robust dual color image blind watermarking scheme in the frequency domain," *Multimedia Tools and Applications*, vol. 83, no. 13, pp. 38711–38735, 2024.
[54] G. Kaur, R. Agarwal, and V. Patidar, "Semi-blind robust watermarking with dual complex tree wavelet based hybrid transform and SVD," in *2020 IEEE 17th India Council International Conference (INDICON)*, 2020, pp. 1–5.
[55] N. Ahmed, H. M. S. Asif, and G. Saleem, "A benchmark for performance evaluation and security assessment of image encryption schemes," *International Journal of Computer Network and Information Security*, vol. 8, no. 12, pp. 18–29, 2016.
[56] K. Deb, M. S. Al-Seraj, M. M. Hoque, and M. I. Hasan Sarkar, "Combined DWT-DCT based digital image watermarking technique for copyright protection," in *2012 7th International Conference on Electrical and Computer Engineering*, 2012, pp. 458–461.
[57] M. A. Provath, K. Deb, P. K. Dhar, and T. Shimamura, "Classification of lung and colon cancer histopathological images using global context attention based convolutional neural network," *IEEE Access*, 2023.
[58] M. Rahman, K. Deb, P. K. Dhar, and T. Shimamura, "ADBNet: An Attention-Guided Deep Broad Convolutional Neural Network for the Classification of Breast Cancer Histopathology Images," *IEEE Access*, 2024.
[59] S. Hossain, K. Deb, S. Sakib, and I. H. Sarker, "A hybrid deep learning framework for daily living human activity recognition with cluster-based video summarization," *Multimedia Tools and Applications*, pp. 1–54, 2024.
[60] J. Talreja, S. Aramvith, and T. Onoye, "DANS: Deep attention network for single image super-resolution," in *IEEE Access*, 2023.
[61] J. Talreja, S. Aramvith, and T. Onoye, "DHTCUN: Deep Hybrid Transformer CNN U Network for Single-Image Super-Resolution," in *IEEE Access*, 2024.

MD. FAHIM HOSSAIN SAIKAT was born in the year 2000 in the city of Kurigram, Bangladesh. He obtained a B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering & Technology (CUET), Chattogram, Bangladesh, in 2024. His research interests include digital watermarking, machine learning, deep learning, and computer vision.

MD. AL-MAMUN PROVATH was born in Jamalpur, Bangladesh, in 1999. He obtained a B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering & Technology (CUET), Chattogram, Bangladesh, in 2023, and is presently pursuing an M.Sc. degree in the same field. He is presently a Lecturer in the Department of Computer Science and Engineering (CSE) at Chittagong University of Engineering & Technology (CUET), Chattogram, Bangladesh. His present research interests encompass image classification, image reconstruction, image enhancement, machine learning, deep learning, data science, and computer vision.

KAUSHIK DEB obtained the B.Tech. and M.Tech. degrees from the Department of Computer Science and Engineering at Tula State University, Tula, Russia, in 1999 and 2000, respectively, and the Ph.D. degree in electrical engineering and information systems from the University of Ulsan, Ulsan, South Korea, in 2011. Since 2001, he has served as a Faculty Member in the Department of Computer Science and Engineering (CSE) at Chittagong University of Engineering & Technology (CUET), Chattogram, Bangladesh. He is presently a Professor at the Department of Computer Science and Engineering at CUET. Additionally, he held multiple administrative roles at CUET, including Dean of the Faculty of Electrical and Computer Engineering (ECE) from 2017 to 2019, Director of the Institute of Information and Communication Technology (IICT) from 2015 to 2017, and Head of the CSE Department from 2012 to 2015. He made several contributions to the management and organization of conferences, seminars, and other academic events. He has authored over 135 peer-reviewed technical articles. His research interests encompass computer vision, deep learning, pattern recognition, intelligent transportation systems (ITSs), and human-computer interaction. He served as a Steering Member. He served as Chair or Secretary at numerous international and national conferences, including the International Conference on Electrical, Computer, and Communication Engineering (ECCE), the International Forum on Strategic Technology (IFOST), the International Workshops on Human System Interactions (HSI), and the National Conference on Intelligent Computing and Information Technology (NCICIT).

PRANAB KUMAR DHAR received his B.Sc. degree from Chittagong University of Engineering & Technology (CUET), Chattogram, Bangladesh, in 2004, his M.Sc. degree from the University of Ulsan, Republic of Korea, in 2010, and his Ph.D. degree from Saitama University, Japan, in 2014. In 2005, he started his role as a Lecturer in the Department of Computer Science and Engineering at CUET, where he currently holds the position of Professor. He is presently a Visiting Scholar at Waseda University in Japan. He has authored two books, one book chapter, and one patent. He has authored over 80 peer-reviewed academic papers and conference proceedings. His research interests encompass multimedia security, digital watermarking, steganography, multimedia data compression, sound synthesis, digital image processing, and digital signal processing. He is on the technical committee of many international conferences. In 2015, he received the Best Paper Award from the Journal of Signal Processing. He was awarded the Best Paper Award at the International Conference on Intelligent Computing (ICIC-2009) in 2009, the Excellent Paper Award at the Engineering and Arts Society Conference in South Korea (EASKO-2009) in 2009, and the Best Paper Award at the 9th International Forum on Strategic Technologies (IFOST-2014) in 2014. He functions as a Reviewer for several esteemed publications, including IEEE, IEICE, Elsevier, and Springer.

TETSUYA SHIMAMURA obtained the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 1986, 1988, and 1991, respectively. In 1991, he became a Professor at Saitama University in Saitama, Japan, where he remains now. He served as a visiting researcher at Loughborough University, U.K. in 1995 and at Queen's University of Belfast, U.K. in 1996. Professor Shimamura has authored or co-authored six books. He is an editorial member of many international publications and a participant in the organizing and program committees of several international conferences. His study focuses on digital signal processing and its applications in voice, picture, and communication systems.